Analysis of cargo stability in container transportation

António José Galrão Ramos

Faculdade de Engenharia da Universidade do Porto
Department of Industrial Engineering and Management

Supervisor
Professor José Fernando da Costa Oliveira
Co-supervisors
Professor José Fernando Gonçalves
Professor Manuel Joaquim Pereira Lopes

Submitted in partial fulfilment of the requirements for the degree of
Doctor of Philosophy in Industrial Engineering and Management.

Faculdade de Engenharia da Universidade do Porto
2015

This research was partially supported by the ERDF through the COMPETE Programme, by the Portuguese Government through the FCT (StableCargo - PTDC/SEN-TRA/121715/2010 and PTDC/EGE-GES/117692/2010) and by the Project BEST CASE - SAESCTN-PIIC&DT/1/2011, which is co-financed by the North Portugal Regional Operational Programme (ON.2 - O Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF).

It is never too late to be what you might have been.
George Eliot

Matthew 25:14-30

To Vanessa, the love of my life, Simão and Teresa.
For your unconditional love.

Acknowledgements

This thesis was a long and challenging endeavour that would not have been possible without the academic and personal support of a number of people, to whom I am very grateful.

I would like to start by thanking Professor José Fernando Oliveira, who generously accepted to be my supervisor, and welcomed me into his research family when I arrived at a crossroads in life. He provided the idea and funding for this research and guided my development as a researcher. Sometimes we just have the good fortune of coming across remarkable people. To him I am forever grateful.

To Professor José Fernando Gonçalves who accepted to co-supervise this work, for his helpful and dedicated supervision throughout the project. His advice and incentive along the way have been an invaluable contribution to this thesis. To Professor Pereira Lopes, co-supervisor, colleague and friend that encouraged and helped me find the right path in my academic career.

To Professor Miguel Gomes, for the brainstorming sessions, patience and companionship. Along with my supervisors', his advice has been of tremendous benefit to the work presented in this thesis. To João Pinho for the support and help with my graphic work. To Lúcia Flores, for all the support with the English language. To Elsa Silva and Pedro Rocha, for their companionship throughout this journey. To Professor Francisco Parreño who kindly provided his CLP solutions to be used as test instances. Special thanks go to the co-authors who have contributed directly to the written parts of this thesis.

In addition, I express my gratitude to the Fundação para a Ciência e Tecnologia and the Instituto Superior de Engenharia do Porto for partially supporting this research.

Finally, to my family, to whom I express my deepest and sincerest love. To my mother and father, who always encouraged and actively supported me all my life, to find and realize my potential. To my in-laws and my sister-in-law for always providing the much needed backup.

To my children, Simão and Teresa, that came into my life during this period. You suffered my absence through the long working hours and travels, but were a fundamental source of motivation and unconditional love. I love both of you - your smiles, your laughs and your powerful hugs.

And also, last but never least, to my loving wife Vanessa, who endured the most hardship from this project with me and enjoyed very few of the joys and excitements in return. You have been my bedrock, providing the stable background that made all of this possible with your love, strength and determination. Being found by you still is my greatest achievement.

Abstract

The main focus of this thesis is to address cargo stability within the container loading problem (CLP). In the last years the transport industry has been facing increasing challenges. Whether as a result of new regulations or of the increasing service levels demanded by customers, the pressure for high levels of performance from transportation companies will continue in the short, as well as in the long term.

This combination between compliance and service level challenges, has a strong implication in the way cargo transportation will be addressed in the future. The need to maximize the usage of space within the different transport modes without deterioration of the cargo is one of the main problems that highlights the effect of these challenges. Given the complexity of the problem, the need for cargo planning tools that can be effectively used in practice, is expected to increase.

The amount of research that has addressed the problem of optimization of the spatial arrangement of cargo inside trucks or containers has grown in the last decade. The problem, known in the literature as the CLP, belongs to the more generic combinatorial optimization class of Cutting and Packing problems.

Although several approaches to this problem have been proposed, their wide adoption by transportation companies did not occur. This mainly results from the fact that the basic formulation of the problem does not address the requirements of the real-world problem, that is, it does not address a set of practical constraints that influence the way cargo is arranged inside the container. One of the most important practical constraints is cargo stability, and even though it has been addressed in the literature, it has been done in an over-simplified way that does not actually translate real-world stability.

The proposed work aims to develop new algorithms for the CLP, based on heuristics and metaheuristics that will contribute to close the gap between research and the industry practice. Particular attention was given to cargo stability, which will be studied, modelled and integrated with the optimization of cargo spatial arrangement.

The first part of this thesis contains a comprehensive introduction to the CLP and cargo stability. This includes CLP typologies, modelling approaches, heuristic solution methods and practical constraints. An extensive analysis of cargo stability within the CLP is also presented.

The second part of this thesis reflects the defined approach to cargo stability, that is, that cargo stability should be addressed by separating static and dynamic stability. It first proposes an algorithm for evaluating static stability based on the static mechanical

equilibrium conditions applied to rigid bodies and a physical packing sequence algorithm that, given a container loading arrangement, generates the actual sequence by which each box is placed inside the container, considering static stability and loading operations efficiency. Secondly, it proposes for the CLP, a multi-population biased random-key genetic algorithm that combines a genetic algorithm and a constructive heuristic algorithm. The constructive heuristic uses the proposed static stability approach to evaluate stability during the filling of the container and the proposed physical packing sequence algorithm to evaluate the feasibility of loading arrangements. Thirdly, to explore dynamic stability, a physics simulation tool is presented and its main physical parameters validated. The tool was developed to simulate the external forces to which the container is exposed during transportation. It is followed by the development of a set of dynamic stability evaluation metrics to be used within CLP algorithms.

Aside from a comprehensive introduction to CLP and cargo stability, the main contributions of this thesis are the proposed static stability approach, the CLP algorithm, the physics simulation tool and a set of dynamic stability metrics.

Keywords: container loading problem, stability, physics engine

Resumo

O principal foco desta tese é a estabilidade de carga no âmbito do problema de empacotamento em contentores (PEC). Nos últimos anos, o sector dos transportes tem vindo a enfrentar desafios cada vez maiores. A pressão sobre as transportadoras para atingir elevados níveis de performance, resultado de novos regulamentos ou dos níveis de serviço exigidos pelos consumidores, continuará, a curto e a longo prazo, a aumentar.

Esta combinação entre cumprimento dos regulamentos e resposta aos desafios ao nível da qualidade de serviço, está intimamente relacionada com a evolução do transporte de carga. A necessidade de maximizar o espaço útil em diferentes meios de transporte, garantindo a integridade da carga, é uma das principais questões enfrentadas.

A investigação realizada na área da optimização espacial de carga em camiões ou contentores aumentou significativamente na última década. O problema, referido na literatura como o PEC, pertence a um conjunto de problemas de optimização combinatória: os Problemas de Cortes e Empacotamentos.

Apesar de já terem sido propostas várias abordagens ao problema, nunca houve uma adopção generalizada por parte das empresas transportadoras, principalmente, porque a formulação base do problema não contempla os requisitos reais, ou seja, não tem em consideração um conjunto de restrições práticas que influenciam a forma como a carga é disposta no interior de um contentor. Uma das restrições mais importantes é a estabilidade da carga, que na literatura, tem sido sempre estudada numa óptica simplista, que não traduz a realidade.

O trabalho proposto pretende desenvolver novos algoritmos para o PEC, baseados em heurísticas e meta-heurísticas, contribuindo para aproximar a investigação e a prática na indústria. Deu-se particular atenção à estabilidade da carga, que será estudada, modelada e integrada com a optimização do arranjo espacial da carga.

A primeira parte desta tese contém uma introdução abrangente ao PEC e à estabilidade da carga, incluindo tipologia do PEC, modelos abordados, métodos heurísticos e restrições práticas. É apresentada uma análise extensiva da estabilidade da carga, no âmbito do PEC.

A segunda parte desta tese reflecte a opção de abordagem à estabilidade da carga, ou seja, que a estabilidade da carga deverá ser analisada separando a estabilidade estática da estabilidade dinâmica.

Em primeiro lugar, propõe um algoritmo para avaliação da estabilidade estática, baseado nas condições de equilíbrio estático de corpos rígidos e um algoritmo de sequenciamento

do carregamento que, a partir da estrutura de carga de um contentor, gera a sequência pela qual cada caixa deverá ser colocado dentro do mesmo, tendo em conta a estabilidade estática e a eficiência das operações de carga.

Em segundo lugar propõe um algoritmo genético multi-população com chaves aleatórias enviesadas para o PEC, que combina um algoritmo genético com uma heurística construtiva. A heurística construtiva utiliza a abordagem da estabilidade estática para avaliar a estabilidade durante a carga do contentor, e o algoritmo de sequenciamento do carregamento proposto para validar a aplicabilidade das sequências de carga.

Em terceiro lugar, é apresentada uma ferramenta de simulação de física, para exploração da estabilidade dinâmica e são validados os seus principais parâmetros físicos. A ferramenta foi desenvolvida para simular as forças externas a que o contentor é sujeito durante o transporte. Segue-se o desenvolvimento de um conjunto de métricas de avaliação da estabilidade dinâmica, a utilizar no contexto dos algoritmos do PEC. Para além de uma introdução exaustiva ao PEC e à estabilidade da carga, as principais contribuições desta tese são a abordagem proposta à estabilidade estática, o algoritmo do PEC, a ferramenta de simulação de física e um conjunto de métricas de avaliação da estabilidade dinâmica.

Palavras-chave : problema de empacotamento em contentores, estabilidade, motor de física

Preface

This thesis has been submitted at the Department of Industrial Engineering and Management, Faculty of Engineering, University of Porto in partial fulfilment of the requirements for acquiring the degree of Doctor of Philosophy in Industrial Engineering and Management.

The PhD study has been partly supported by the ERDF via the COMPETE Program, by the Portuguese Government through the FCT (StableCargo - PTDC/SENTRA/121715/2010 and PTDC/EGE-GES/117692/2010) and by the Project BEST CASE - SAESCTN-PIIC&DT/1/2011, which is co-financed by the North Portugal Regional Operational Programme (ON.2 - O Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), from July 2012 to January 2015.

The work was supervised by Full Professor José Fernando da Costa Oliveira from the Department of Industrial Engineering and Management, Faculty of Engineering, University of Porto and co-supervised by Associate Professor José Fernando Gonçalves, from the Faculty of Economics, University of Porto and Associate Professor Manuel Joaquim Pereira Lopes from the School of Engineering, Polytechnic of Porto.

The thesis deals with different aspects of optimisation within the container loading problem and consists of two parts. The first part serves as an general introduction to the container loading problem and to the specific topic of cargo stability explored during the PhD study. This part also contains an overview of the work conducted during the PhD study as well as an overall conclusion on the conducted study. The second part of the thesis contains a collection of four research papers written during the PhD study. These research papers are all co-authored. One of the four papers is in press and available on-line in a peer review journal and the other three are currently submitted to peer reviewed journals within the field.

Contents

List of Tables

List of Figures

Part I

Cargo stability within the container loading problem

Chapter 1

Introduction

This thesis addresses cargo stability within the container loading problem (CLP). The CLP is a combinatorial optimization problem, which belongs to the wider combinatorial optimization class of Cutting and Packing problems (C&P). As frequently occurs in operations research problems, it has a multidisciplinary nature that encompasses other areas such as computer science and mechanical engineering.

The CLP as it has been addressed in the field of operations research has found limited applicability in practice, mainly due to a limited or inefficient approach to the problem when taking into account a number of practical constraints, such as the container weight limit, weight distribution or cargo stability.

It is this framework that is in the genesis of the research program, and the project from which it was designed, "StableCargo - Cargo stability analysis in container transportation: a hybrid optimization - heuristics framework" funded by the Fundação para a Ciência e Tecnologia and contracted by INESC TEC. The aim of the project was to address the CLP, a real world driven combinatorial optimization problem with particular focus given to cargo stability.

The expected outcomes of the StableCargo project were:

- New stability measures that take into account both loading stability and transportation stability;
- Implementation of a software model to access cargo plans stability measures;
- Improved container loading algorithms, that incorporate stability goals in the solution build and search processes;
- A simulation tool, based on a physics dynamics model, which will allow the simulation of cargo transportation and constitute a workbench for new stability measures and criteria development.

To achieve the project goals, the research program followed a hierarchical approach by tackling first the challenges related with the CLP loading stability and then moving to the CLP transportation stability.

The work presented in this thesis is divided into two parts. The first part (which includes this introduction), describes the CLP and reviews the literature on CLP with particular focus on the stability constraint. The second part of the thesis contains four scientific papers that reflect the research done throughout the course of this study.

This chapter introduces the background and motivation for the research project, details its aims and objectives and provides an overview of each chapter of the thesis.

1.1 Background and motivation

Cargo transportation has played an important role throughout the history of mankind. Either by the proximity to the sea or by the control of land routes, the connection to the exterior as a way to expand and create wealth, was at the genesis of the great civilizations of history. It is not by chance that most of those civilizations of ancient times, developed around the Mediterranean Sea and the large rivers that pay tribute, taking the advantages offered by river and maritime transportation.

This link between transports and the development of mankind has been present throughout history until the modern day. The effect of globalization led to a world where products and services are exchanged in increasing numbers and distances by and to an increasing number of origins and destinations.

The role that cargo transportation plays as the basis of worldwide trade, places enormous challenges to transport managers to continuously research for ways to increase the rational use of transportation. Cost, security, speed, efficiency or consistency in transport are permanently under scrutiny.

Cargo transportation management has the goal of transporting the correct product, in the correct quantity, to the correct location at the lowest cost, in the correct instant and the desired quality. It covers a broad spectrum of fields that go beyond the transport mode selection. It includes issues like route definition, cargo packaging and conditioning, handling, insurance, etc.

Packaging takes an important role in the support of the logistical process. It has an impact on the majority of logistics activities since the size and density of packaging are directly linked with transport and warehousing cost (Ballou, 1999).

The purpose of packaging goods for transportation is to protect them from possible damage, allow for proper handling, and obtain an efficient space usage inside the transport unit while permitting the stability of the transport vehicle (Hellström and Saghir, 2007).

Throughout the several levels of packaging, the unitization of the product is obtained, contributing to a more efficient and safe handling and transportation of the product.

One of the major innovations within the field of transport management in the last 100 years was the container. The shipping container was invented and patented in 1956 and has revolutionized the transport of goods and cargo worldwide and has made containerization the standard method of transporting break bulk cargo. Containerization of cargo contributes to the speedy loading and unloading of cargo as well as providing additional protection for the cargo from theft, breakage, and contamination. From a logistics per-

spective it conducted to a "unitarization" of loads, that is, loads with the same dimensions that permitted multimodal transportation (Levinson, 2008).

It was expected that through containerization, the cargo damage would be limited. The reality showed that putting cargo in a metal box does not guarantee that it will arrive in good condition. Figure 1.1 sets out the most frequent types of container cargo damage that give rise to major insurance claims, measured against all cargo damage. According to the financial insurer UK P&I Club, 20% of all container cargo related claims can be traced back to bad stowage (UK P&I Club, 2000).

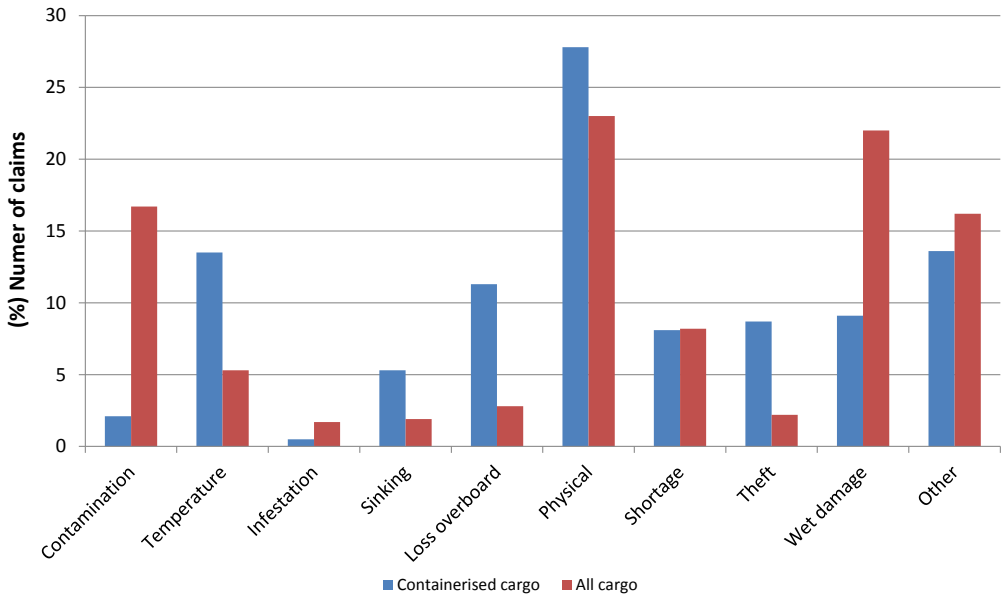


Figure 1.1: Large cargo claims - Type of damage (UK P&I Club, 2000)

For safety reasons, it is of the utmost importance that when cargo is placed on the transport vehicle, it is done in a way that it can neither endanger persons nor goods and cannot slide or fall off the vehicle. Particularly on road transport, safety during cargo transportation is a major concern. According to the European Union, cargo-securement failure contributes to up to 25% of accidents involving trucks (EC, 2012). Important efforts were directed towards developing guidelines and standards for cargo securing. Table 1.1 presents the main guidelines and standards.

Table 1.1: Guidelines and standards for cargo securing

Document	References
IMO/ILO/UN ECE Guidelines for Packing of Cargo Transport Units	(IMO/ILO/UNECE, 2014)
European Best Practice Guidelines on Cargo Securing for Road Transport	EC (2010)
Container Handbook	GDV (2003)
EN standard 12195-1:2010 - Load restraint assemblies on road vehicles	ECS (2010)

Cargo safety is not the only challenge in transportation management. The European Commission considers that an efficient transport system is a condition for maintaining prosperity in the EU, and the present scenario of oil dependency, growing congestion and looming climate change, requires a new strategy for the European transportation systems. The strategy established aims to achieve a 60% reduction of the CO_2 emissions and of oil dependency by 2050, without limiting the freedom of movement. A strong focus is placed in the optimising the performance of multimodal logistic chains, and on increasing the efficiency of transport with information systems (EC, 2011).

It is thus expected that in the next decades, freight transportation costs will rise, and the need for increased efficiency and security in the transportation sector will drive companies to adopt cargo planning tools that can be effectively used in practice. It is not thus surprising the amount of research that has been carried out in the operations research field, that focus on the optimization of the spatial arrangement of cargo inside transportation vehicles or containers, maximizing the space usage of cargo transport units (CTU).

This problem is known in the literature as the CLP, which belongs to the more generic combinatorial optimization class of Cutting and Packing problems. In the CLP, a set of rectangular shaped boxes (small items) must be packed orthogonally in a set of containers (three dimensional, rectangular large objects), in a way that the boxes do not overlap and all the boxes of the subset lie entirely within the container, as to maximize the space usage (Wäscher et al., 2007).

The majority of CLP approaches, found in the literature, are of limited applicability in practical situations. The existence of a series of real world constraints that have a strong influence on the loading arrangement of a container, that are not properly addressed, contributes to that limitation (Bischoff et al., 1995; Bortfeldt and Gehring, 2001). The main constraints encountered in practice are (Bischoff et al., 1995; Bortfeldt and Wäscher, 2013):

- Cargo stability - refers to the necessity of guaranteeing the integrity of the cargo, as well as safety conditions on loading, unloading and transport operations;
- Complexity of loading arrangements - refers to the effort required to handle materials in order to fulfil loading patterns;
- Complete shipment of certain groups of items - addresses the need for cargo grouping as a result of product functional or administrative reasons;
- Grouping of items - refers to the impact of grouping cargo as a way to improve loading and unloading operation efficiency;
- Positioning of items within a container - refers to the impact that the interrelationships between the properties of products during transport can have on the cargo location and to the direct impact that the weight or size of an item can have on the positioning of the item in a container. This impact can also be due to the handling and loading equipment required;

- Shipment priorities - refers to the priority shipping rating of an item, as a result, for example, of delivery deadlines;
- Orientation - refers to the orientation instruction for box positioning. It can be the result of, for example, the "this side up" instruction on a box or the pallet loading orientation;
- Load bearing strength of items - refers to the amount of weight a box can support per unit area. The instruction on a box "Stack no more than x items high" illustrates this constraint;
- Container weight limit - refers to the maximum weight that can be loaded inside a container. For high mass density products, the main constraint is the container weight limit, not the container volume;
- Weight distribution within the container - refers to the the distribution of the weight of the cargo along and across the container. It affects the handling and transportation.

The stability constraint previously referred has already been studied in the literature. Previous works that address the cargo stability constraint uses the term "stability" in a rather simplified way and sometimes as if the term was self-explanatory. Different concepts, as stability during loading operations and stability during transportation are not addressed separately and are dealt with as no different. Usually, stability is achieved by guaranteeing that each box has a full base support or almost full base support. This approach penalizes the space usage of the container as the existing cargo stability measures only partially grasp the stability problem essence, failing to assess the way in which real loading stability and transportation stability are achieved (Bischoff et al., 1995).

The present research aims to refine the concept of cargo stability and develop an evaluation of cargo stability under a realistic framework. It is considered that cargo stability has been addressed in the past under over simplified assumptions, and that cargo stability must be studied separating static from dynamic stability. This goal is pursued by developing stability metrics that can be applied to solutions generated by existing CLP algorithms, or used within algorithms, and to develop an efficient algorithm for the CLP that incorporates the new stability approach.

1.2 Problem description

The container loading problem can be stated as follows:

A given set of n types of small items of parallelepiped shape, designated as boxes, $B = (b_1, b_2, ..., b_n)$ where for each box type $i = (1, ..., n)$ the following set of characteristics is defined:

- length (d_i);

- width (w_i);
- height (h_i);
- quantity (q_i);
- all boxes have the same density;
- all boxes are perfectly rigid;
- the centre of gravity of each box is assumed to be at its geometric centre;
- all boxes have the same friction coefficient,

are to be loaded in a large object of parallelepiped shape, designated as container C with the following characteristics:

- length (D);
- width (W);
- height (H);
- the length, width and height lie parallel to the x, y and z-axis of the first octant of a Cartesian coordinates system, with the back-bottom-left corner lying in the coordinates system origin;
- the container provides lateral support;
- with rigid walls;

with the objective of achieving a maximum utilization of the volume of the container, while meeting the following loading constrains:

- orthogonal loading of each box - each face of the box must be parallel to the faces of the container;
- no overlapping between the boxes;
- all the boxes lie entirely within the container;
- the packing of the boxes into the container is constrained by the physical packing sequence, that is, from back to front;
- all the walls of the container are solid,

and the practical constraint:

- stability constraint - the ability of each box to maintain the loading position without significant change. Stability must be considered during:
  - the cargo loading operation (usually addressed in the literature as vertical or static stability) and;
  - the transport operation (usually addressed in the literature as horizontal or dynamic stability).

The placement of a box b_{ij} , that is, the j^{th} box of type i, in a loading space is given by its minimum and maximum coordinates, $(x1_{ij}, y1_{ij}, z1_{ij})$ and $(x2_{ij}, y2_{ij}, z2_{ij})$, respectively.

Even though these assumptions reduce the complexity of the problem, it remains extremely hard to solve.

In the typology proposed by Wäscher et al. (2007) for Cutting and Packing problems, the problem addressed in this work can be classified as the 3D Single Knapsack Problem (SKP) or the 3D Single Large Object Placement Problem (SLOPP), depending on the level of heterogeneity of the cargo, with additional stability constrains.

1.3 Thesis structure

This thesis is structured in two separate parts. The first part contains an introduction to the CLP and the specific topics explored in this PhD study. Aside from this introductory chapter, Part I consists of three separate chapters:

- Chapter 2 describes the main characteristics of the CLP, reviews the literature on the CLP with the stability constraint and discusses in further detail the most commonly used and interesting solution methods for this problem.
- Chapter 3 lists the main contributions of the work conducted during this PhD study. This includes a detailed overview of the work contained in the research papers that constitutes Part II of this thesis.
- Finally, Chapter 4 expounds the conclusion on the work of this thesis. It also highlights the contributions from this thesis and reflects on possible directions for future research within the CLP.

The second part of this thesis contains the four research papers written during the PhD study that constitute the major scientific contributions of this thesis:

- Chapter 5 addresses the stability of the cargo during container loading and proposes two algorithms. The first is a Static Stability Algorithm based on static mechanical equilibrium conditions that can be used as a stability evaluation function embedded in CLP algorithms (e.g. constructive heuristics, metaheuristics). The second proposed algorithm is a Physical Packing Sequence Algorithm that, given a container loading arrangement, generates the actual sequence by which each box is placed inside the container, considering static stability and loading operations efficiency constraints.

- Chapter 6 explores the use of the static stability constraint based on the static mechanical equilibrium conditions applied to rigid bodies which derive from Newton's laws of motion embedded in a constructive heuristic placement procedure, used as part of a multi-population biased random-key genetic algorithm. The new hybrid genetic algorithm is extensively tested on well-known benchmark instances using three variants: no stability constraint, the classical full base support constraint and with the new static mechanical equilibrium stability constraint—a comparison is then made with the state-of-the-art algorithms for the CLP. The results show that with the new static stability criterion, a higher percentage of space utilization is obtained on average, than with the classical full base support condition. Moreover, for highly heterogeneous cargo the new hybrid genetic algorithm with full base support constraint outperforms the other literature approaches.

- Chapter 7 proposes a physics simulation tool to be used for evaluation of dynamic stability of CLP solutions within a realistic framework. The proposed physics simulation tool is based on a physics engine and a library for computer graphics based on OpenGL. The results of the physics simulation tool are compared with the state-of-the-art simulation engineering software Abaqus Unified FEA, having allowed us to conclude that our tool is a promising alternative for dynamic stability evaluation.

- Chapter 8 addresses dynamic stability within the CLP. It starts by proposing two new performance indicators to evaluate the dynamic stability of cargo arrangements, the number of fallen boxes (NFB) and the number of boxes within the Damage Boundary Curve damage area (NB_DBC). Using 1500 solutions generated by a CLP algorithm for well known literature problem instances, the performance indicators are measured using the physics simulation tool, StableCargo, presented on Chapter 7. Then, the existing literature metrics for dynamic stability, that is, the mean number of boxes by which items other than those on the floor are supported (M1) and the percentage of boxes with insufficient lateral support (M2), and the percentage of volume occupied in the container, are evaluated by measuring the correlation between the metrics and performance indicators results. Since the calculation of the NFB and the NB_DBC performance indicators using StableCargo is computationally expensive, two new dynamic stability metrics are proposed, so that they can be integrated within a container loading algorithm. The metrics are models of the proposed dynamic stability performance indicators, derived using multiple linear regression analysis. The results show that the proposed metrics are a better surrogate measure for dynamic stability measurement for the CLP.

At the time of writing, four of the research papers have been submitted to international journals, one is in press and available on-line and three are in the peer revision process.

The layout of the papers follows the style of the remainder of the thesis, nevertheless, the content of each paper is exactly the same as the content submitted to the journals.

Chapter 2

State of the art

The purpose of this chapter is to present the current understanding and conceptualization of the cargo stability constraint within the context of the CLP, which influences the direction that the research will take and the areas of the literature that the dissertation will focus on. It starts by focusing on the positioning of the CLP within the Cutting and Packing problems, the different solution approaches to the problem and the main constraints found in the scientific literature. It will then focus on the different approaches to the stability constraint and how it is addressed in the literature.

2.1 Types of Container Loading Problem

The Container Loading Problem is a combinatorial optimization problem, which belongs to the wider combinatorial optimization class of the Cutting and Packing problems. The essential form of the Cutting and Packing problems can be summarised as follows: given a set of large objects and a set of small items (both defined in a number of geometric dimensions), the small items must be assigned to the large objects and a dimensional objective function is optimised satisfying two geometric conditions, all small items lie entirely within the large objects and the small items do not overlap (Wäscher et al., 2007).

A preliminary analysis of the literature on Cutting and Packing problems led to the conclusion that it has received a lot of attention, and that the problems it addresses are referred to by many names in literature. This is due to the large spectrum of industrial applications that these problems address and to the different fields in which they are inserted. The multitude of Cutting and Packing problems names, which several times refer to the same type of problem, posed difficulties in investigating the underlying structure within Cutting and Packing problems and in facilitating the cross-fertilisation of research within the academic community.

In order to classify the Cutting and Packing problems, Dyckhoff (1990) proposed a typology that could describe the type of Cutting and Packing problems based on four characteristics. However, the typology proposed by Dyckhoff (1990) has not been widely accepted. Wäscher et al. (2007) identified several limitations of the Dyckhoff (1990) typol-

ogy and presented an improved typology, which modified the Dyckhoff (1990) criteria for the definition of problem types. The criteria used by the two typologies are presented in Table 2.1.

The combination of the different criteria from the Wäscher et al. (2007) typology, allows to distinguish different problem types. Figures 2.1 and 2.2 present the problem types with input minimization objective and output maximization objective, respectively.

		assortment of small items	
characteristics of large objects		**weakly heterogeneous**	**strongly heterogeneous**
all dimensions fixed	**identical**	Single Stock-Size Cutting Stock Problem (SSSCSP)	Single Bin-Size Bin Packing Problem (SBSBPP)
	weakly heterogeneous	Multiple Stock-Size Cutting Stock Problem (MSSCSP)	Multiple Bin-Size Bin Packing Problem (MBSBPP)
	strongly heterogeneous	Residual Cutting Stock Problem (RCSP)	Residual Bin Packing Problem (RBPP)
one large object variable dimensions(s)		Open Dimension Problem (ODP)	
		(ODP/W)	(ODP/S)

Figure 2.1: Input minimization problem types (Wäscher et al., 2007)

		assortment of small items		
characteristics of large objects		**identical**	**weakly heterogeneous**	**strongly heterogeneous**
all dimensions fixed	**one large object**	Identical Item Packing Problem (IPP)	Single Large Object Placement Problem (SLOPP)	Single Knapsack Problem (SKP)
	identical		Multiple Identical Large Object Placement Problem (MILOPP)	Multiple Identical Knapsack Problem (MIKP)
	heterogeneous		Multiple Heterogeneous Large Object Placement Problem (MHLOPP)	Multiple Heterogeneous Knapsack Problem (MHKP)

Figure 2.2: Output maximization problem types (Wäscher et al., 2007)

The general definition of the CLP can be stated as a three dimensional (3D) Cutting and Packing Problem where a set of parallelepiped shaped boxes (regular small items) must be packed orthogonally in a set of parallelepiped shaped containers (regular large objects), in a way that the boxes do not overlap and all the boxes of the subset lie entirely

Table 2.1: Cutting and Packing typologies

Dyckhoff (1990) typology	Wäscher et al. (2007) typology
Dimensionality	
• one dimensional • two dimensional • three dimensional • N dimensional with $N > 3$	• one dimensional • two dimensional • three dimensional
Kind of assignment	
• all objects and a selection of items • a selection of objects and all items	• output value maximization • input value minimization
Assortment of large objects	
• one object • identical figure • different figures	• one large object – all dimensions fixed – one or more variable dimensions – several large objects • identical large objects – weakly heterogeneous assortment – strongly heterogeneous assortment
Assortment of small objects	
• few items (of different figures) • many items of many different figures • many items of relatively few different (non-congruent) figures • congruent figures	• identical small items • weakly heterogeneous assortment • strongly heterogeneous assortment
Shape of small items (for two and three dimensional problems)	
	• regular small items • irregular small items

within the container. As an assignment problem it can have two basic objectives, the output value maximization (knapsack problem) and the input value minimization (bin packing problem). The first one refers to problems where the number of containers is not sufficient to accommodate all the boxes. The latter refers to problems where the number of containers is sufficient to accommodate all the boxes (Wäscher et al., 2007).

This thesis addresses two types of problems with an output value maximization objective, with one large object with fixed dimensions and regular shaped small items. These problems can be classified either as three-dimensional, rectangular single large object placement problems (3D-SLOPP) or as three-dimensional, rectangular single knapsack problems (3D-SKP), depending on the cargo heterogeneity. This class of problems is the focus of the thesis and the remaining literature review will mainly focus on these problems.

2.2 Modelling approaches

The CLP is considered an extremely difficult problem. That is probably the main reason why in the literature, only a small number of models and exact methods were proposed. The proposed models for the SKP and SLOPP problems are mainly mixed integer linear (MIP) models. One of the first models was proposed by Fekete and Schepers (1997). They presented a general framework for the exact solution of the multi-dimensional knapsack packing problems using an approach based on graph-theoretical characterization of feasible packings. Fasano (1999) proposed an MIP model, based on the one that Chen et al. (1995) proposed for the CLP problem types with an input minimization objective, where variables are based on a Cartesian coordinate system. As additional constraints Fasano (1999) includes the *static balancing* constraint, that is, the weight distribution constraint defined by Bortfeldt et al. (2003). Padberg (2000) later extended the model of Fasano (1999) to achieve a minimal formulation of the problem. Allen et al. (2012) extended the Fasano/Padberg model by adding further constraints, but concluded that even though it achieved a time reduction, the formulation was still of limited applicability to solve instances with more than 20 boxes. Considering the results obtained, Allen et al. (2012) proposed an alternative formulation that uses space-indexed binary variables, that place the boxes in a discrete container where its units of space are given by the largest common denominator of the respective dimensions of the boxes. This approach presented better performance than the one of Padberg (2000). Junqueira et al. (2012b) also proposed an MIP model with variables based on a Cartesian coordinate system that included constraints for the vertical and horizontal stability of the cargo, and for its load bearing strength. An extension to this model that includes a multidrop constraint was proposed by Junqueira et al. (2012a), and an integration with the Vehicle Routing Problem formulation was proposed by Junqueira et al. (2013). All of these proposals are only suitable to solve small-sized instances.

2.3 Heuristic algorithms

The usual approach to the CLP is based on heuristics, that allow achieving a near-optimum solution within an acceptable amount of time. Even though exact methods have had a considerable development, good solutions for real-world problem sized instances can only be obtained in a reasonable amount of time by heuristic algorithms. Fanslau and Bortfeldt (2010) proposed a categorization of heuristic methods. They were categorized according to the method type, as conventional heuristics, metaheuristics and tree-search methods.

Conventional heuristics include construction heuristics and improvement heuristics. Construction heuristics derive a single feasible solution that is directly returned or used as a starting point for local search heuristics. They do not attempt to improve the obtained solution. Improvement heuristics try to iteratively improve already known solutions (Rothlauf, 2011). Examples include the construction heuristics developed by George and Robinson (1980), Bischoff et al. (1995) and Lim et al. (2003) and the improvement heuristic of Egeblad et al. (2010). Metaheuristics can be seen as general-purpose methods that aim to effectively and efficiently explore a search space using intensification (exploitation) and diversification (exploration) strategies. According to the philosophy followed, metaheuristics can be seen as extended variants of improvement heuristics that aim to escape from a local optimum solution and continue with the exploration of the search space with the expectation of finding a better solution, or as a population-based approach where the search space is explored in each iteration by a population (Blum and Roli, 2003). Examples of such approaches can be found in the Genetic Algorithms of Bortfeldt and Gehring (2001) and Gonçalves and Resende (2012), in the use of Tabu Search by Bortfeldt et al. (2003) and Liu et al. (2011a) and the greedy randomized adaptative search procedures (GRASP) of Moura and Oliveira (2005) and Parreño et al. (2008). The *Tree-search methods* include tree-search and graph-search methods. These are methods that can be used when the set of all feasible solutions of the optimization problem can be represented by a tree or a graph. Examples include the works of Fanslau and Bortfeldt (2010), Zhu and Lim (2012) and Araya and Riff (2014). Metaheuristics and Tree-search methods search over a representation or codification of the solution (usually box sequences) and therefore require a constructive heuristic for box loading in order to generate a feasible solution to the problem.

The vast number of packing heuristics found in literature use different approaches and methodologies which influences the structure of generated packing plans. In order to differentiate the approaches Pisinger (2002) suggests a classification based on four loading patterns of the container: *wall-building*, *stack-building*, *guillotine-cutting* and *cuboid-arrangement*. Fanslau and Bortfeldt (2010) later extended this classification by dividing the loading patterns as *wall-building*, *stack-building*, *horizontal layer-building*, *block-building*, and *guillotine-cutting*.

The *wall-building* approach fills the container with vertical layers ("walls") transversal to the depth of the container. Figure 2.3 presents an example of the *wall-building* approach.

The *stack-building* approach fills the container with stacks. Each box is packed in suitable stacks which are then arranged on the floor of the container by solving a two-

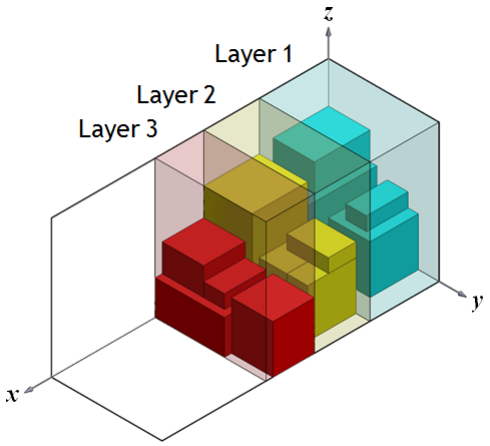


Figure 2.3: Illustration of a *wall-building* approach

dimensional packing problem. The stacks do not form walls as defined before. In Figure 2.4 an example of the *stack-building* approach is presented.

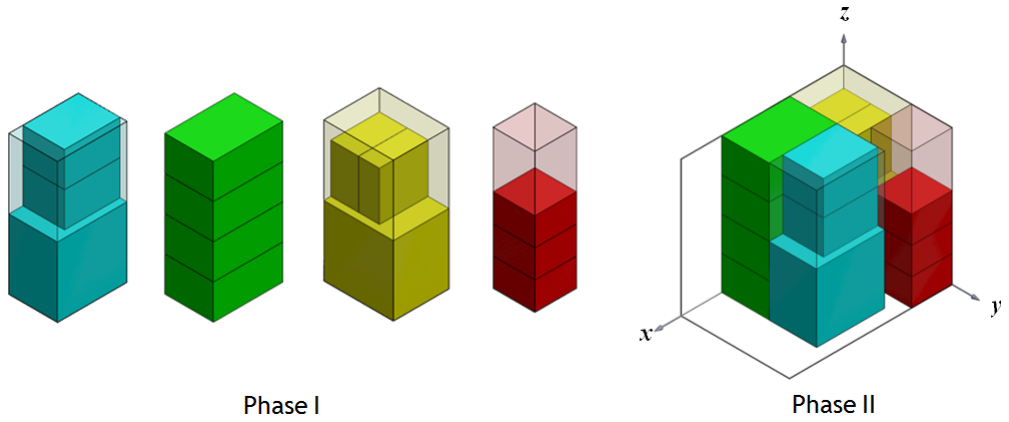


Figure 2.4: Illustration of a *stack-building* approach

The *horizontal layer-building* approach fills the container using horizontal layers from the bottom to the top of the container. It is similar to the *wall-building* approach and is based on the pallet loading problem layer approach. An example is presented in Figure 2.5.

The *block-building* approach fills the container with cuboid blocks. Cuboid blocks are arrangements of similar boxes with the same spatial orientation. An illustration of the *block-building* approach is presented in Figure 2.6.

The *guillotine-cutting* approach is based on a slicing tree representation of a packing plan. Each slicing tree corresponds to a successive segmentation of the container into smaller pieces by means of guillotine cuts, where the leaves correspond to the boxes to be packed. A *guillotine-cutting* approach example is presented in Figure 2.7.

Even though the proposed classification of the packing heuristics approaches of Pisinger (2002) and Fanslau and Bortfeldt (2010) help to enlighten the different approaches, they

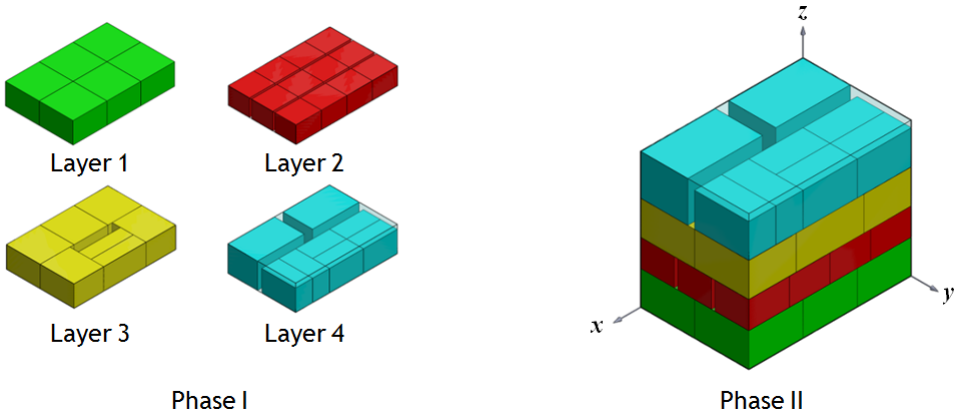


Figure 2.5: Illustration of a *horizontal layer-building* approach

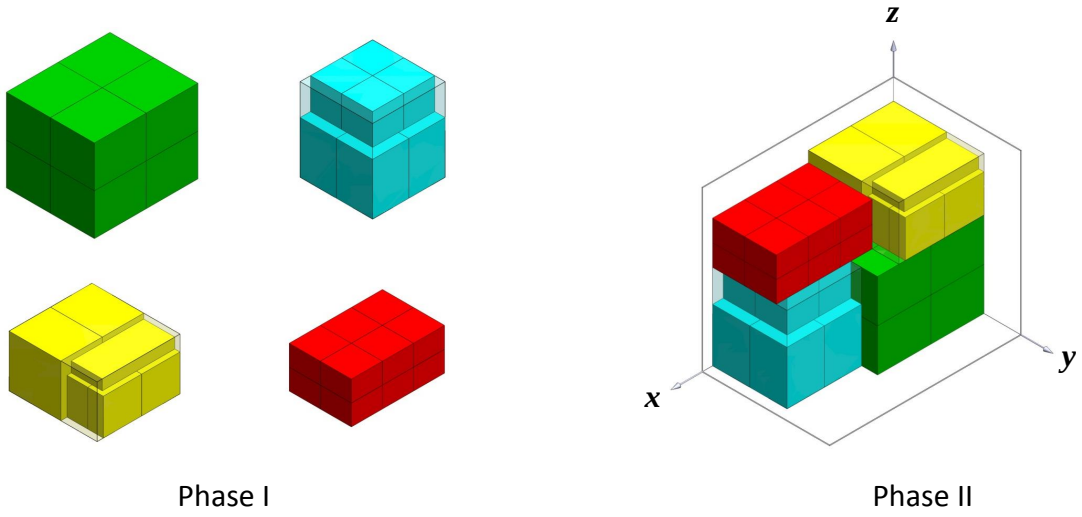


Figure 2.6: Illustration of a *block-building* approach

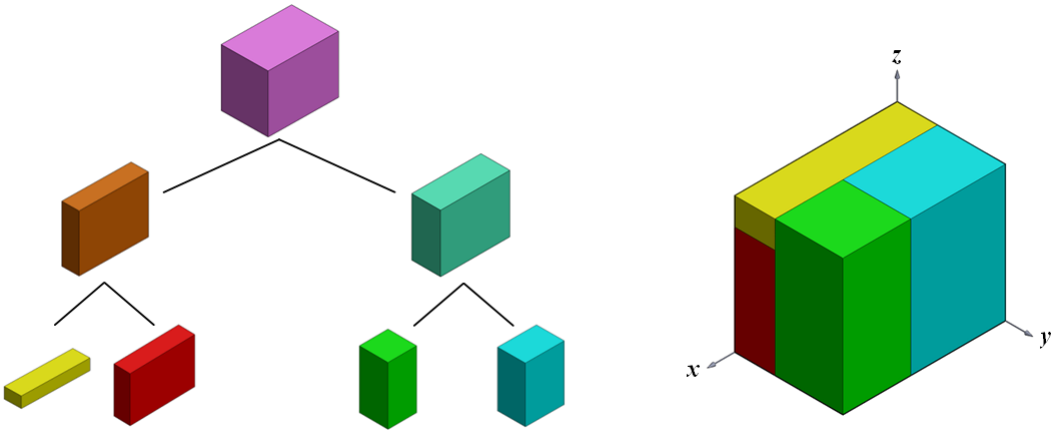


Figure 2.7: Illustration of a *guillotine-cutting* approach

do not actually translate the fundamental structural decisions of the authors in those approaches.

Usually, the constructive heuristic iteratively selects a location inside the container and a box (or set of boxes) to place at that location, until no more locations or boxes are available. Both these decisions are related to the way the empty space of the container is managed (and therefore the way in which all potential placement locations are evaluated) and the way box arrangements are generated. The constructive heuristics result from the combination of the selected spatial representation and box arrangement strategy and a set of rules that select location and box arrangement at each iteration. Considering these elements we present the most commonly used spatial representation of the container and the box arrangement strategies.

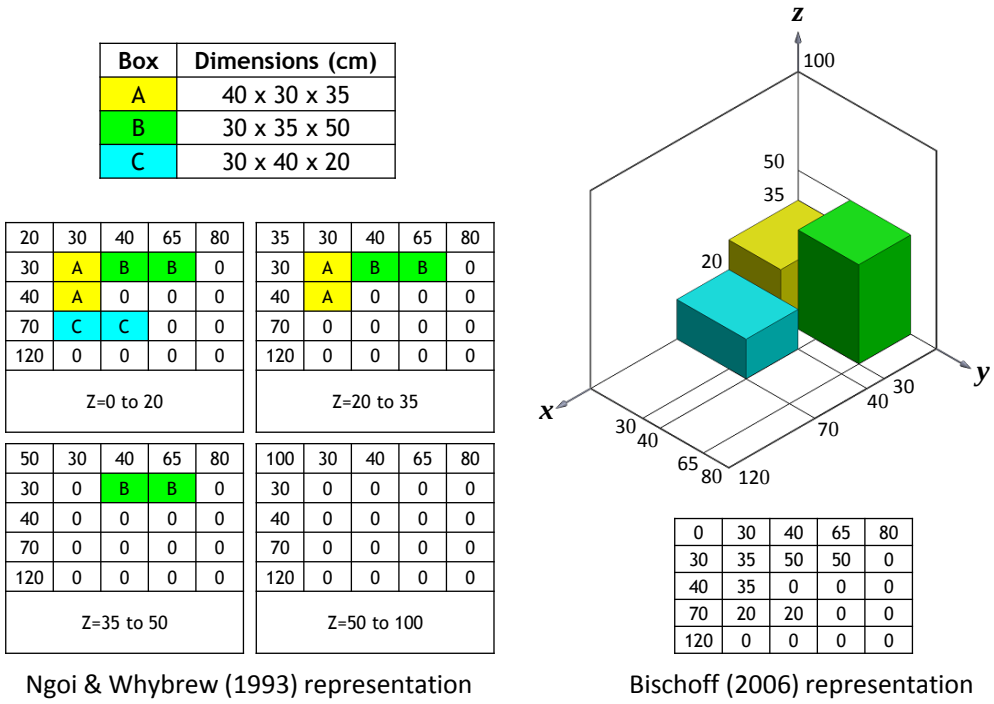
2.3.1 Spatial representation

The spatial representation method used to manage the space inside the container is an element of relevance and distinction, in the different approaches to the CLP, and influences the identification of potential placement locations of boxes.

Ngoi and Whybrew (1993) use a single three-dimensional matrix representation of objects and empty spaces as a combination of variable orthorhombic cells. Each three-dimensional matrix is composed by a chain of two-dimensional matrices that represent the details of horizontal layers of constant thickness. In each layer of the matrix the first row contains the y coordinates of the vertices of the boxes (excluding the first position), the first column contains the x coordinates (excluding the first position) and the row one, column one cell contains the z coordinate of the layer. The remaining matrix cells contain the identification number of a packed box or the value 0 (zero) if the space is empty. Bischoff (2006) proposed an adaptation of the Ngoi and Whybrew (1993) representation, that does not involve the creation of the horizontal layer matrices. Instead a single two dimensional matrix that represents a view from the top of the container is required. The first row and first column of the matrix have the same meaning as in the Ngoi and Whybrew (1993) method, but the contents of the remaining cells correspond to the height of potential loading surfaces. An example of a three box representation is shown in Figure 2.8.

The concept of *envelopes* is also used for space management in the CLP, and is appropriate for methods that place one box at a time. The purpose of the *envelope* is to divide the container volume into two sub-volumes, an inaccessible and an accessible one, and to determine a set of *corner points* that define the positions where a new box can be placed. The *corner points* are the points of the envelope were its slope changes from vertical to horizontal. The *envelope* and *corner points* are updated each time a box is placed in the container. This approach, illustrated in Figure 2.9 was firstly used in the CLP by Martello et al. (2000).

Even though this method is efficient and accurate, it may result in direct loss of space, because it can encapsulate volume in the inaccessible region that cannot be used later on. To solve this problem, Crainic et al. (2008), developed the concept of *extreme points*, that is, a series of potential insertion points generated each time a box is placed, that include the



Box	Dimensions (cm)
A	40 x 30 x 35
B	30 x 35 x 50
C	30 x 40 x 20

20	30	40	65	80
30	A	B	B	0
40	A	0	0	0
70	C	C	0	0
120	0	0	0	0

Z=0 to 20

35	30	40	65	80
30	A	B	B	0
40	A	0	0	0
70	0	0	0	0
120	0	0	0	0

Z=20 to 35

50	30	40	65	80
30	0	B	B	0
40	0	0	0	0
70	0	0	0	0
120	0	0	0	0

Z=35 to 50

100	30	40	65	80
30	0	0	0	0
40	0	0	0	0
70	0	0	0	0
120	0	0	0	0

Z=50 to 100

Ngoi & Whybrew (1993) representation

0	30	40	65	80
30	35	50	50	0
40	35	0	0	0
70	20	20	0	0
120	0	0	0	0

Bischoff (2006) representation

Figure 2.8: Representation of three boxes in a container

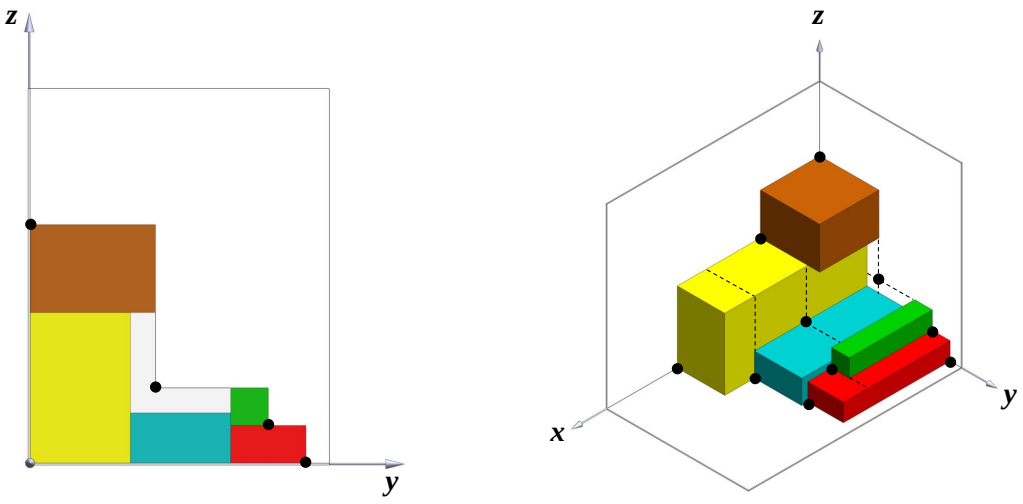


Figure 2.9: Example of *corner points* in 2D and 3D packings

corner points of Martello et al. (2000) and insertion points for the otherwise inaccessible region (see Figure 2.10).

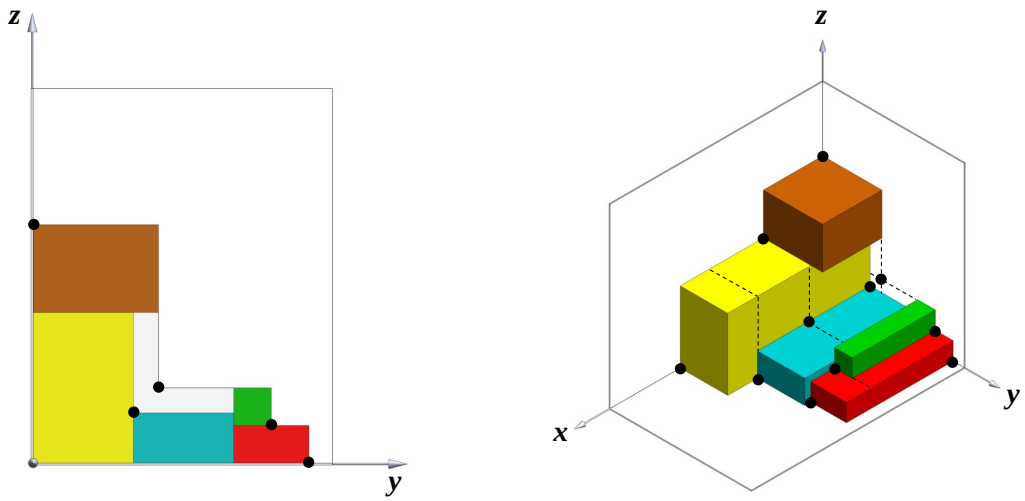


Figure 2.10: Example of *extreme points* in 2D and 3D packings

Tao and Wang (2013) had a similar approach to Crainic et al. (2008) to overcome potential loss of volume, but they distinguished between *corner points* and *normal points*. In Figure 2.11 the approach is illustrated.

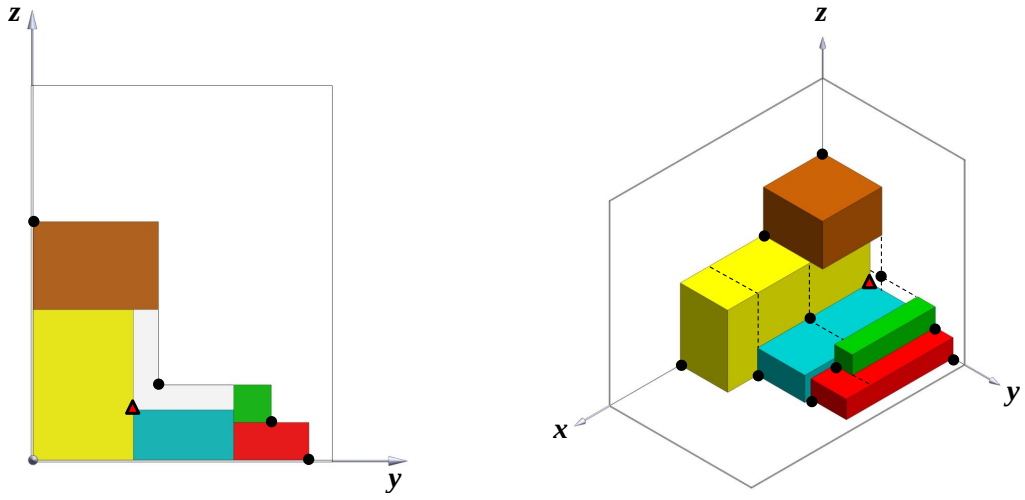


Figure 2.11: Example of *corner points* and *normal points* in 2D and 3D packings

Another approach to space management was proposed by George and Robinson (1980). Their methodology is based on the assumption that after the placement of a box in a packing space, the remaining unused space gives origin to three new spaces. The three spaces, illustrated in Figure 2.12, are created in the following order: spare depthwise space,

spare widthwise space and spare heightwise space. Each space is therefore represented by its length, height and width and the coordinates of its rear left bottom vertex.

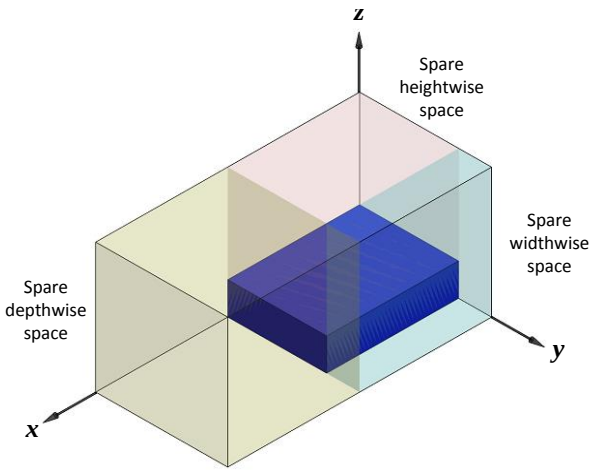


Figure 2.12: George and Robinson (1980) empty space subdivision

This approach only considers one way of partitioning the empty space. Other authors, like Bortfeldt et al. (2003) and Fanslau and Bortfeldt (2010) later extended the George and Robinson (1980) approach and considered other subdivision variants of the empty space, as illustrated in Figure 2.13. In these approaches empty spaces are represented as a set of disjoint spaces.

For the two and three-dimensional cutting stock problem, Lai and Chan (1997) proposed a representation of empty spaces as a set of non-disjoint empty spaces. These empty spaces have the largest parallelepiped shape that can be considered and are managed using the "Interval Generation" procedure. However, this procedure was not applied to the three-dimensional CLP. Later Parreño et al. (2008) used this representation in the CLP and designated the non-disjoint empty spaces as *maximal-spaces*. The *maximal-spaces* representation is illustrated in Figure 2.14. A *maximal-space* s representation is given by its minimum and maximum coordinates, (x_{1s}, y_{1s}, z_{1s}) and (x_{2s}, y_{2s}, z_{2s}), respectively, and an Insertion Vertex V, (i.e., the vertex of the *maximal-space* where the boxes will be packed).

2.3.2 Box arrangement strategy

A box can be placed inside the container in at most 6 different orientations (Figure 2.15, depending on its orientation constraints. The strategy followed when placing boxes inside the container is also a key element to differentiate the approaches proposed in the literature. We can separate between single box or multiple box strategies: in each iteration of the former, only one box is placed inside the container; in each iteration of the latter a set of boxes are placed together.

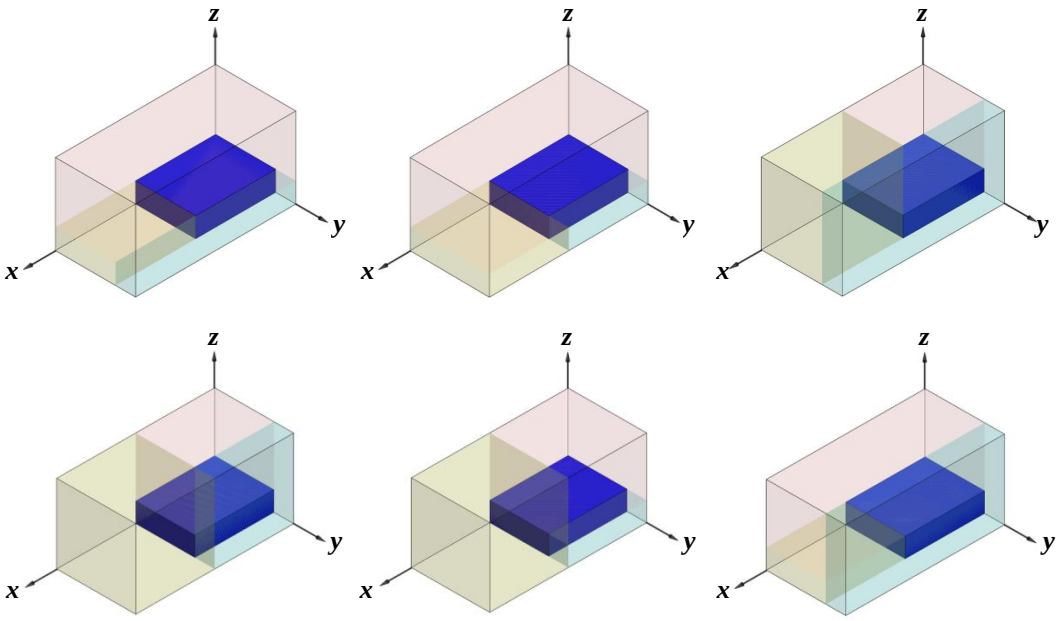


Figure 2.13: Variants of the empty space subdivision

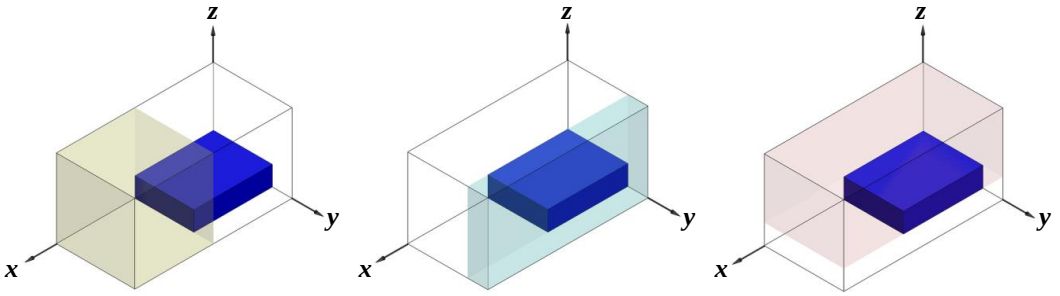


Figure 2.14: Maximal-space representation

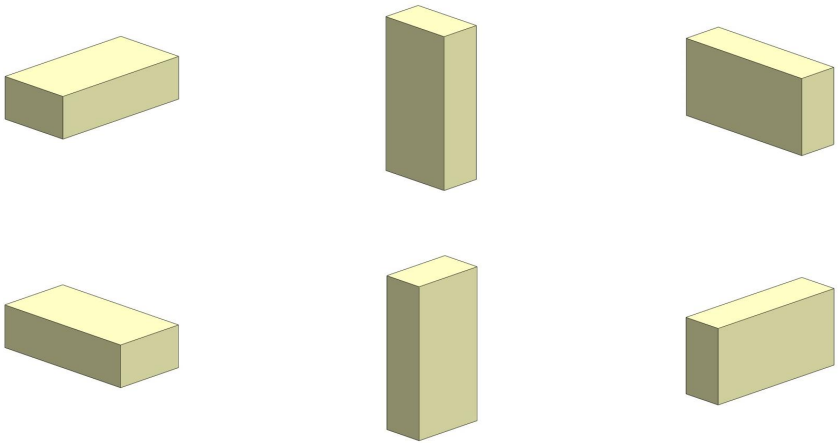


Figure 2.15: Six different alternatives for positioning a box

In a multiple box strategy the arrangements can be formed from sets of identical or non-identical boxes. Multiple boxes can also be arranged by dimensionality, that is, one, two and three dimensions. In the one dimensional arrangement, boxes are grouped along a single axis, and therefore generate a column. Usually the dimension of the column is determined by the empty space available, and only identical boxes with the same orientation are considered. Figure 2.16 illustrates, for a defined orientation of a box, the three alternatives for generating a column of identical boxes. Examples can be found in Parreño et al. (2008).

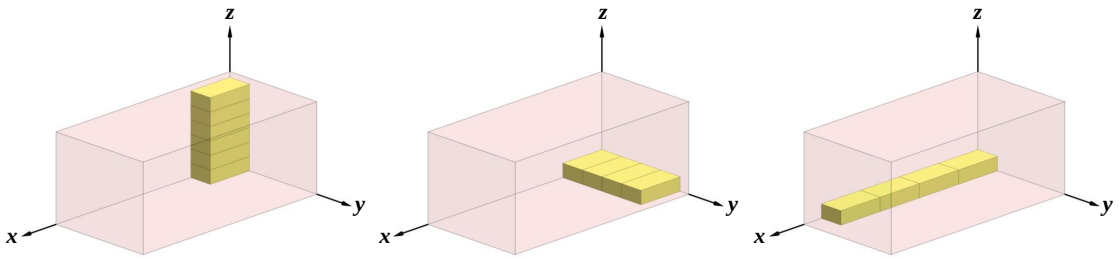


Figure 2.16: Three different alternatives to generate a column

In a two dimensional arrangement, boxes are grouped in a plane, that is, along two axes, generating layers. In this case, the layer is built first by generating a column along one of the axis of the plane, and then replicating the column along the other axis of the layer. It is the available space and the first axis selected for layer building that most frequently determines the layer configuration. The two dimensional arrangement is usually determined for identical boxes with the same orientation (Figure 2.17). Examples can be found in Parreño et al. (2008) and Gonçalves and Resende (2012). Both one and two dimensional approaches have as main purpose to build faster algorithms while taking advantage of the box identical dimensions for better space usage.

In the three dimensional arrangement, boxes are grouped along the three axes, generating blocks. A high number of approaches fall into this category. Whether designated as stacks, walls, or blocks, these approaches have the main purpose of diminishing the complexity of the problem through the grouping of boxes. In the existing approaches two types of blocks can be found, simple and general blocks (Fanslau and Bortfeldt, 2010). In simple blocks all the boxes are identical and have the same orientation. In general blocks there can be several types of boxes and different orientations of identical boxes. Figure 2.18 illustrates the two types of blocks. Examples can be found in Fanslau and Bortfeldt (2010), Zhu and Lim (2012) and Araya and Riff (2014).

The most effective CLP algorithm in the literature to date that uses a two dimensional arrangement strategy is the multi-population biased random-key genetic algorithm developed by Gonçalves and Resende (2012), while the most effective CLP algorithm that uses a three dimensional arrangement strategy is the Beam Search Approach developed by Araya and Riff (2014).

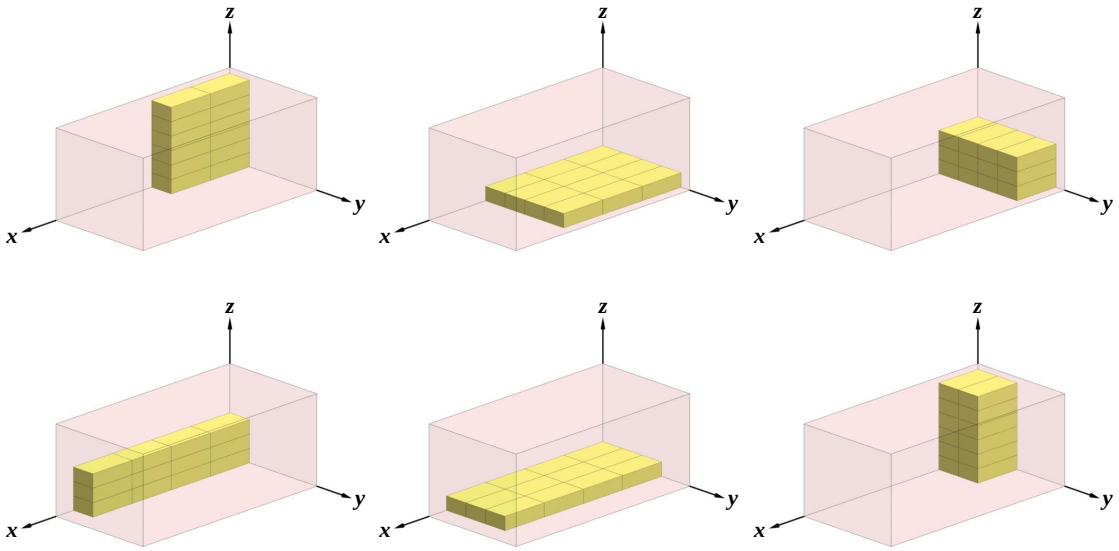


Figure 2.17: Six different alternatives to generate a layer

(a) Simple block (b) General block

Figure 2.18: Different alternatives to generate a block

2.4 Benchmark instances

The purpose of benchmark instances is to compare the performance of different algorithms in order to identify state-of-the-art approaches. There are only a few sets of instances for the CLP, and from these, only a few consider an additional constraint, which is the orientation constraint. For problem types with input minimization objective there are four main data sets:

- The Ivancic et al. (1989) data set contains 47 instances and is used in the SSSCSP problem;
- The Martello et al. (2000) data set contains 320 instances that include orientation constraint. It is for the SBSBPP problem;
- The Mack et al. (2004) data set that contains 100 instances that include orientation constraints. Used for the ODP/W problem;

- and the Bortfeldt and Gehring (1999) data set that contains 100 instances that include orientation constraints. It is for the ODP/W and ODP/S problems.

As for problem types with output maximization objective there are three main data sets:

- The Loh and Nee (1992) data set contains 15 instances and is used in the SLOPP problem;
- The Bischoff et al. (1995) data set contains 700 instances that include orientation constraints. It is used for the SLOPP problem;
- and the Davies and Bischoff (1999) data set that contains 800 instances that include orientation constraints. It is used for the SKP problem.

The most commonly used data sets do not incorporate characteristics of the container and/or boxes that allow the comparison of CLP algorithms taking into account a higher number of constraints. However, there are other data sets, such as the data set of Ratcliff and Bischoff (1998), that include orientation constraint, the weight and the load bearing capacity of the boxes.

When considering the CLP with output maximization objective, approximately 50% of the papers use the Bischoff et al. (1995) 700 test instances. Loh and Nee (1992) and Davies and Bischoff (1999) test instances are also used in about 25% of the papers. It must be referred that in the more recent approaches to the CLP problem types that consider the output maximization objective, the SLOPP and SKP problems are also tested using the data set instances of Bischoff et al. (1995) and Davies and Bischoff (1999). These problem instances are organized in 15 classes, with a total of 100 instances per class. They are designated as BR1 to BR15. The instances used cover a wide range of situations. The heterogeneity of the boxes increases from just 3 different box types in BR1 to 100 box types in BR15. The average number of boxes per box type also varies from 50.15 boxes per type in BR1 to 1.33 in BR15. The dimensions of the boxes are generated independently from the dimensions of the container and the total volume of the boxes in each individual instance never exceeds the container volume. On average the total volume of boxes to be packed represents 99.46% of the container volume (Table 2.2).

2.5 Practical constraints

As previously mentioned in Chapter 1 there are a number of relevant practical constraints that should be taken into account when tackling the CLP due to the influence they have in cargo arrangements.

The designations of practical constraints found in the literature can sometimes be confusing. Expressions like loading stability (Abdou and Yang, 1994), stack stability (Bischoff, 1991), load bearing support (Bischoff, 2006), stacking constraint (Bortfeldt et al., 2003), balance constraints (Bortfeldt and Gehring, 2001), load bearing (Christensen and Rousøe,

Table 2.2: Characteristics of the classes of problem instances of Bischoff et al. (1995) and Davies and Bischoff (1999)

Class	Number of problems	Type of Boxes	Average Number of Boxes	Average Volume boxes/ Container volume
BR1	100	3	150.44	99.58
BR2	100	5	136.65	99.47
BR3	100	8	134.3	99.49
BR4	100	10	132.85	99.43
BR5	100	12	132.87	99.36
BR6	100	15	131.47	99.45
BR7	100	20	130.33	99.37
BR8	100	30	130.66	99.52
BR9	100	40	128.89	99.40
BR10	100	50	130.16	99.54
BR11	100	60	129.47	99.47
BR12	100	70	130.31	99.48
BR13	100	80	130.41	99.50
BR14	100	90	129.96	99.47
BR15	100	100	129.88	99.44

2009), stable loading (Egeblad and Pisinger, 2009), fragility constraints (Gendreau et al., 2006), etc. can be interrelated with one another or be referring to the same practical constraint.

It is also observed that in the literature, in the majority of cases, practical constraints are not properly defined since they are explained mainly through their implementation in the problem. Nevertheless, some authors provide a definition for the terms used. For example, Abdou and Yang (1994) define "Loading stability" as "the ability of a box to maintain its position", Bischoff (2006) defines "load bearing" as "the ability of items to withstand pressure from the weight resting on them".

Using the work of Bischoff et al. (1995) as a starting point, Bortfeldt and Wäscher (2013) performed an analysis of the CLP literature and suggested a classification for the practical constraints. They distinguish between:

- container-related constraints;
  - weight limit - it is related to the maximum weight that can be loaded inside a container;
  - weight distribution - it addresses the weight distribution of the cargo across the container. It is related with the centre of gravity of the container.
- box-related constraints;
  - loading priorities - is related to the degree of importance of the loading of a box.
  - orientation - is related to the orientation of the box.

  - stacking - is related to the weight that a box can withstand without deforming.

- cargo-related constraints;
  - complete shipments - reflects the need of a set of boxes to be packed together.
  - allocation - expresses the need to pack a set of boxes into the same container.

- positioning constraints - restricts the absolute position of a box and/or the relative position of boxes inside the container.

- load-related constraints;
  - stability - it is related to the stability of the cargo.
  - complexity - it is related to the difficulty of implementation of the cargo arrangement.

The classification proposed by Bortfeldt and Wäscher (2013) translates the way the practical constraints have been addressed in the literature and is very useful. However, it does not provide a general structure that allows to establish a relation between constraints. For example, the weight distribution constraint is related with the stability of the cargo during transportation and nevertheless they are not in the same category of constraints.

A new classification structure that provides a base that relates the practical constraints was deemed necessary. Based on the identified constraints in the literature, they were classified into two main types: safety constraints and logistics constraints. Safety constraints are related to the integrity of the cargo and the transport unit, and the safety of workers and external parties during loading and transport operations. Logistics constraints address operational decisions that are not related with the physical properties of cargo and transport unit. Table 2.3 presents the classification of the existing constraints according to the classification proposed here.

Table 2.3: Practical constraints classification

Safety constraints	Logistics constraints
weight limit	loading priorities
weight distribution	complete shipments
orientation	allocation
stacking	complexity
positioning	positioning
stability	

Safety constraints are strongly interrelated. The physical properties of the container and boxes influence the way they withstand all stresses to which they are subjected during handling and transportation. Table 2.4 presents the main box properties that influence the different safety constraints.

Table 2.4: Box properties required for safety constraints

Safety constraints	Mechanical Properties				Biological Properties	Chemical Properties
	Density	Center of mass	Friction coefficient	Load bearing strength		
weight limit	x					
weight distribution	x	x				
orientation	x	x				
stacking	x			x		
stability	x	x	x	x		
positioning					x	x

2.5.1 Cargo Stability in the Container Loading Problem

This thesis focuses on the cargo stability constraint. It is therefore of the utmost importance to establish the meaning of what is considered as cargo stability and an evaluation of the approaches to stability in the CLP literature.

It is considered that cargo stability is the ability of each box to maintain the loading position without significant change during cargo loading and transportation operations. Stability during loading operations is usually addressed in the literature as vertical or static stability while stability during the transport operation is usually addressed in the literature as horizontal or dynamic stability (Bortfeldt and Wäscher, 2013).

The paper of Bortfeldt and Wäscher (2013) on CLP constraints was used as the starting point for the literature review on cargo stability in the CLP. Bortfeldt and Wäscher (2013) reviewed CLP articles that focused on CLP and practical constraints, which were published or made available on-line in leading academic journals, edited volumes, and conference proceedings between 1980 and the end of 2011. The same criteria were used and the time frame extended to the end of 2014, but the search was restricted to publications dealing with 3D Cutting and Packing problems that included additional cargo stability related constraints, considering the typology proposed by Wäscher et al. (2007). This means that not only 3D Container Loading, but also 3D Pallet Loading Problems were taken into account.

It is, however, important to refer that there are a couple of relevant differences between the two problems. In the CLP the walls of the container provide lateral support to the cargo, while in the Pallet Loading Problem the lateral support is not present (Bischoff and Ratcliff, 1995a). Another difference between these problems concerns the physical plane loading pattern, that is, pallet loading usually follows an horizontal loading pattern instead of the usual vertical loading pattern of container loading (Abdou and Yang, 1994).

As of December 2014, 61 papers have been identified, that focused on CLP and cargo stability. Table A.1 and Table A.2 in the appendix list these papers. The papers were classified according to the following criteria:

Table A.1

- Problem Type - according to the Wäscher et al. (2007) typology;

- Assortment of large objects
- Type of large object - container or pallet
- Assortment of small items
- Type of heuristic method - according to Fanslau and Bortfeldt (2010);
- Test instances - Test instances used in the CLP.

Table A.2

- Safety Constraints - according to the new proposed classification (based in Bortfeldt and Wäscher (2013));
- Logistics Constraints - according to the new proposed classification (based in Bortfeldt and Wäscher (2013));
- Stability approach;
- Spatial representation strategy;
- Box arrangement strategy;
- Loading patterns of the container - according to Fanslau and Bortfeldt (2010);

Both the problem type and practical constraints classifications were, to a large extent, already done in the Bortfeldt and Wäscher (2013) review. Figure 2.19 presents the number of publications per year on the CLP and Pallet Loading that address cargo stability constraints. It can be observed that in recent years cargo stability constraint has had an increase of interest from literature. It is even considered by some authors as one of the most important constraints in the CLP (Bischoff, 1991; Moura and Oliveira, 2005; Parreño et al., 2008).

Despite the growing interest in cargo stability, it has been treated in a rather over-simplified way by the majority of the authors. Some consider that, since the objective of the CLP is to obtain the maximum use of the container volume, a high volume occupation level would naturally translate into a stable cargo. Some authors also consider that any solution is physically stable through the use of additional supports or filler materials. However, this approach implies an extra cost and volume disposal that it is not taken into account in their algorithms.

The way static stability constraint has been addressed has not changed significantly since it was proposed by Carpenter and Dowsland (1985): by imposing that each box must be fully supported by other boxes or by the container floor, stability is achieved. Dynamic stability is addressed by less than 20% of the papers examined. As in static stability, dynamic stability is approached in an over-simplified way: it is evaluated through the lateral support of each box. The most common condition for considering that a box has sufficient horizontal support is being surrounded by other boxes or the lateral walls

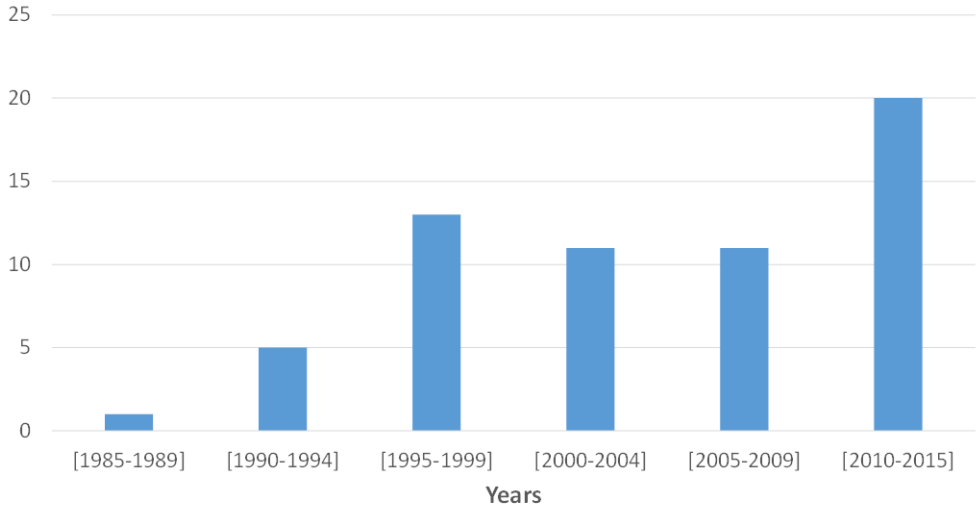


Figure 2.19: Number of publications that address the cargo stability constraint

of the container from at least three sides. These criteria can be deconstructed with a few examples. A pyramid of boxes or a wall of boxes, as illustrated in Figure 2.20, can have full base support or three-sided support but are clearly unstable. The over-simplified approach can also be a reason why stability is not addressed separately from other practical constraints.

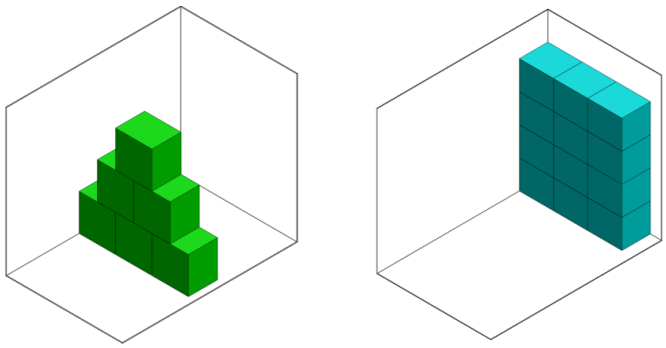


Figure 2.20: Unstable patterns examples with full base support

Static stability

The approaches to static stability found in the CLP literature can be classified according to the type of stability constraint to be enforced: full base support, partial base support or static mechanical equilibrium. Both full base support and static mechanical equilibrium guarantee static stability, while partial support does not.

- **Full base support** requires that the entire base of a box is in contact with the base of the container or with the top surface of other boxes. As a result, no overhanging

boxes are allowed. Examples can be found in Bischoff and Ratcliff (1995a), Bortfeldt and Gehring (2001), Gonçalves and Resende (2012) and Zhu and Lim (2012).

- **Partial base support** requires that either the entire base of a box is in contact with the base of the container, or a pre-specified percentage of the area of a box base is in contact with the top surface of other boxes, thereby allowing overhanging. As an example, Carpenter and Dowsland (1985) requires the contact area to fall in the range of 95% to 75%, while Christensen and Rousøe (2009) require a minimum of 80%, Gendreau et al. (2006), Fuellerer et al. (2010) and Tarantilis et al. (2009), 75%, Gehring and Bortfeldt (1997), 70% and Mack et al. (2004), 55%.

- **Static mechanical equilibrium** requires that the entire base of a box is in contact with the base of the container or,

  – the sum of external forces acting on the box is zero and;

  – the sum of torque exerted by the external forces is zero.

  The first authors to actually mention the static mechanical equilibrium conditions in the context of the CLP are de Castro Silva et al. (2003). However they have only formulated the constraints and did not provide an algorithm to enforce them. Other authors, enforce stability conditions that are related with the static mechanical equilibrium conditions applied to rigid bodies. Lin et al. (2006) enforce the center of gravity condition which requires the center of gravity of a box be located above the contact surface of its supporting boxes. However, by itself, the center of gravity condition does not guarantee static stability. Mack et al. (2004) stability condition requires the center of gravity of a box to be located above the contact surface of its supporting boxes and the supported base of each box, calculated in relation to its supporting boxes and the container floor, to be greater than a predetermined value. Figure 2.21 illustrates these conditions. While the center of gravity is directly above a supporting box, only a small part of its base is supported indirectly by the container floor and therefore it is considered not stable. Even though this condition avoids the generation of some unstable arrangements that could be generated if only the center of gravity condition was enforced, it also prevents the possibility of generating some stable ones.

Recent approaches in the literature consider the concept of static stability as equivalent to enforcing full base support (Gonçalves and Resende, 2012; Zhu and Lim, 2012; Araya and Riff, 2014). Consequently these authors developed both algorithms that enforce full base support and algorithms that do not have such requirement (Unsupported). The performance measure used by these algorithms is the percentage of volume loaded with or without full base support. As a result, the goal is not to obtain a loading arrangement that is statically stable but a loading arrangement where all boxes have full support.

Table 2.5 summarizes the best results of existing CLP approaches with stability constraints enforcement. In the table, Parallel_HYB.XL refers to the parallel hybrid local

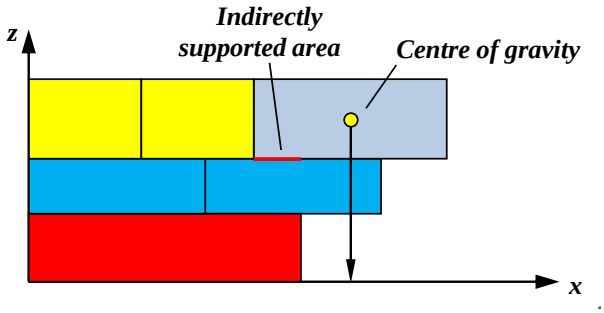


Figure 2.21: Mack et al. (2004) center of gravity condition example

search algorithm of Mack et al. (2004), BRKGA refers to the multi-population biased random-key genetic algorithm of Gonçalves and Resende (2012), HBMLS refers to the block-loading heuristic based on multi-layer search of Zhang et al. (2012) (two versions are presented, the AS version that uses simple blocks and the AC version that uses composite bocks), ID-GLTS refers to the iterative-doubling greedy-lookahead algorithm of Zhu and Lim (2012) and BSG-CLP refers to the Beam Search Approach of Araya and Riff (2014). Presented in the table header are the spatial representation, the box arrangement strategy and the stability criterion used by each of the algorithms. The values from columns 2 to 7 of the table correspond to the average percentage of volume utilization for the test instances of Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999) organized in 15 classes, with a total of 100 instances per class. Classes BR1 to BR7 are considered to be weakly heterogeneous classes while BR8 to BR15 are considered to be strongly heterogeneous. An analysis of the results shows that the full base support stability constraint is the one that is mostly used in the algorithms with the best performance. However it can not be concluded that the full base support constraint outperforms the other stability approaches.

The use of full base support as a stability constraint is very costly for the efficiency of a CLP algorithm, particularly in the strongly heterogeneous instances. Figure 2.22 presents the difference in percentage points between the average results of the CLP algorithms solutions without considering static stability (Unsupported) and the CLP algorithms solutions considering static stability, for each class of problems. It can be observed that in each algorithm the difference follows a similar pattern, that is, for the weakly heterogeneous classes BR1 to BR7 the value of the difference is almost constant, while from classes BR8 to BR15, the strongly heterogeneous, the difference increases with the increase of the heterogeneity of the classes. The average difference of all algorithms for classes BR1 to BR7 is 1.03% and for classes BR8 to BR15 is 3.17%. In the most heterogeneous class BR15 the average difference is above 4%. To give an idea of the impact that this represents on transportation costs, 4% of the volume of a 20-foot container is around 1.35 m^3 of space.

Another element that should be taken into account when addressing static stability is the feasibility of the physical packing sequence. The physical packing sequence is the sequence by which each box is placed inside the container in a specific location determined by a CLP algorithm (Ngoi et al., 1994) and it is closely related with static stability.

Table 2.5: Comparison of the best existing algorithms with static stability constraints

	Parallel_HYB.XL (2004)	BRKGA (2012)	HBMLS (AS) (2012)	HBMLS (AC) (2012)	ID-GLTS (2012)	BSG (2014)
Spatial representation	Empty Spaces	Empty Spaces	Empty Spaces	Empty Spaces	Empty Spaces	Empty Spaces
Box arrangement	3D - Block	2D - Layer	3D - Block	3D - Block	3D - Block	3D - Block
Stability criterion	55% base support	Full Support	Full Support	Full Support	Full Support	Full Support
BR1	93.70	94.34	94.30	93.95	94.40	**94.50**
BR2	94.30	94.88	94.74	94.39	94.85	**95.03**
BR3	94.54	95.05	94.89	94.67	95.10	**95.17**
BR4	94.27	94.75	94.69	94.54	94.81	**94.97**
BR5	93.83	94.58	94.53	94.41	94.52	**94.80**
BR6	93.34	94.39	94.32	94.25	94.33	**94.65**
BR7	92.50	93.74	93.78	93.69	93.59	**94.09**
BR8	-	92.65	92.88	93.13	92.65	**93.15**
BR9	-	91.90	92.07	**92.54**	92.11	92.53
BR10	-	91.28	91.28	92.02	91.60	**92.04**
BR11	-	90.39	90.48	**91.45**	90.54	91.40
BR12	-	89.81	89.65	90.91	90.35	**90.92**
BR13	-	89.27	88.75	90.43	89.69	**90.51**
BR14	-	88.57	87.81	89.80	89.07	**89.93**
BR15	-	87.96	86.94	89.24	88.36	**89.33**
Mean (BR 1-7)	93.78	94.53	94.46	94.27	94.51	**94.74**
Mean (BR 8-15)	-	90.23	89.98	91.19	90.55	**91.22**
Mean (BR 1-15)	-	92.24	92.07	92.63	92.40	**92.87**

∗ **The best values appear in bold**

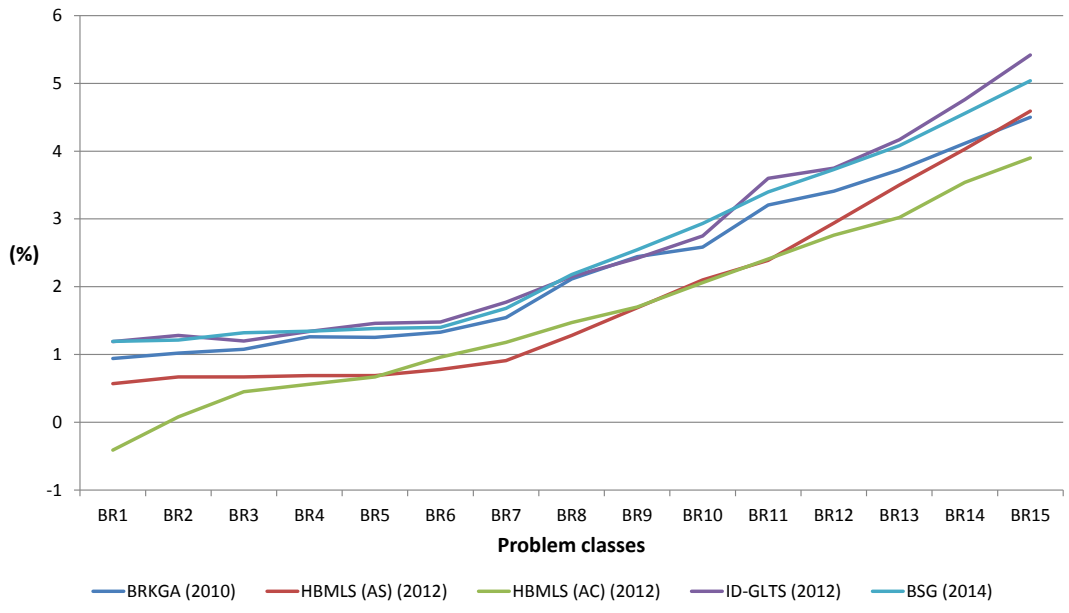


Figure 2.22: Difference in percentage points between the results of the unsupported and static stability variants

It must be observed that the sequence by which solutions are generated by CLP algorithms, that is, the sequence by which the algorithm fills the space, does not necessarily correspond to the actual loading sequence. As such, only when the CLP problem incorporates additional requirements (e.g. multi-drop situations), or the loading sequence is directly related with the nature of the problem, as when the CLP is combined with the VRP, is there the need to address the loading or unloading sequence (Bortfeldt and Wäscher, 2013).

Therefore, the problem of determining the actual packing sequence of boxes has been ignored in the majority of the CLP approaches. However, there are authors that make a distinction between placing sequence and physical packing sequence (Bischoff and Ratcliff, 1995a), and address the generation of the physical packing sequence, by proposing packing in vertical and horizontal layers (Ngoi et al., 1994). Nevertheless, they do not provide any physical packing sequence algorithm (Bischoff and Ratcliff, 1995a; Ngoi et al., 1994). In algorithms that use a back-bottom-left criterion to select the location for inserting a box, and place one box at a time (de Castro Silva et al., 2003), the placing sequence can be a viable physical packing sequence.

Dynamic stability

Dynamic stability has also been tackled in a simplified way in the CLP and only a handful of authors have addressed it. The majority of these approaches address dynamic stability as a soft constraint, by including a set of rules in the algorithms that contribute to improve the results of a set of metrics that represent the degree of dynamic stability of a solution. These metrics however are over-simplified and do not provide a realistic representation of dynamic stability.

While with the static stability constraint there is no significant difference between the CLP and the pallet loading problem, that is not the case with the dynamic stability constraint. The difference lies in the physical limits of the large item. In the container the walls have a high resistance to deformation and can provide an effective lateral support (metal sheets, or side curtains with side boards) that contribute to the dynamic stability of the cargo. The pallet is limited by stretch film that has as main goal to convert a set of boxes into a single rigid body, but has a smaller contribution to the dynamic stability of the cargo.

The approaches found in the CLP literature are classified according to the type of container wall: flexible or rigid wall. A flexible container wall is regarded as not providing lateral support for the boxes, in opposition to rigid walls.

In problems with flexible container walls, authors consider that in order to achieve dynamic stability, the possibility of generating arrangements that allow guillotine cuts must be avoided. Guillotine cuts are considered to contribute to an unstable pallet. As such, authors focus on having boxes interlocking each other to avoid guillotine cuts. Two different approaches to interlocking can be found in the literature. The first, by Carpenter and Dowsland (1985) and Bischoff (1991), treat interlocking as a soft constraint and use a set of criteria to evaluate the dynamic stability of generated layouts, while the second

proposed by Abdou and Elmasry (1999) enforces the interlocking of the column stacks but does not measure it.

Two stability criteria to measure dynamic stability were proposed by Carpenter and Dowsland (1985). One criterion states that each box must have its base in contact with at least two boxes, ignoring contact surface areas with less than a predetermined percentage value of the base of the box, and aims to evaluate the degree of interlocking of the pallet. Another criterion considers the problems related with the pallet guillotine section cutting in the vertical direction. The criterion states that the guillotine cut must not exceed a predetermined percentage value of the maximum length or width of the stack. Bischoff (1991) replaced the first criterion by another where each box positioned on the perimeter of the pattern has to be supported by at least two boxes in the layer below, ignoring contact surfaces areas with less than a predetermined percentage value of the base of the box.

In problems with rigid container walls, all the authors treat dynamic stability as a soft constraint. They consider that it is the interlocking and/or the limited lateral movement of boxes that contribute to dynamic stability. To evaluate interlocking, Bischoff et al. (1995) presented two metrics. The first one is the average number of supporting boxes for each box that is not positioned on the container floor. The higher the value the better. The second is similar to the first but does not consider contact areas with less than 5% of the base area of a box. To evaluate the lateral movement Bischoff et al. (1995) proposed the percentage of boxes that does not have at least three of its four lateral sides in contact with another box or with the container walls. The smaller the value the better. The first interlocking metric (M1) and the lateral movement metric (M2) are the metrics used in the CLP to evaluate dynamic stability.

If all the forces that cargo must withstand during transportation have to be taken into consideration, whether it is by road, railway or sea, the approach that has been followed is effectively over-simplified. Dynamic forces result from the handling and transport operations. During transportation, forces result from events like braking, turning or lifting in land transport or by rolling or vertical movements at sea. Figure 2.23 illustrates the forces acting on the cargo during road transport. The measure of these dynamic forces is usually determined through the acceleration since these forces can be determined by the product of the mass of the cargo and its acceleration. The reference acceleration values for safety evaluation of cargo in different transport modes are presented in Table 2.6 and expressed as a product of the gravitational acceleration ($g = 9.81 m/s^2$).

The impact of these forces on cargo can lead to different types of cargo movement. One of these movements is sliding. Sliding occurs when the force acting on a box overcomes the friction force generated between the box and its supporting surface. Sliding is illustrated in Figure 2.24a.

Another type of movement is tipping. Tipping occurs when a force acting on a box creates a rotational movement of the box. Figure 2.24b illustrates tipping. Sliding and tipping are the basic cargo movements. Other types of movement can occur in flexible loading arrangements. Figure 2.25 presents some examples. These are just a few examples that illustrate that stability is much more complex than the simplified approach generally used when considering the CLP.

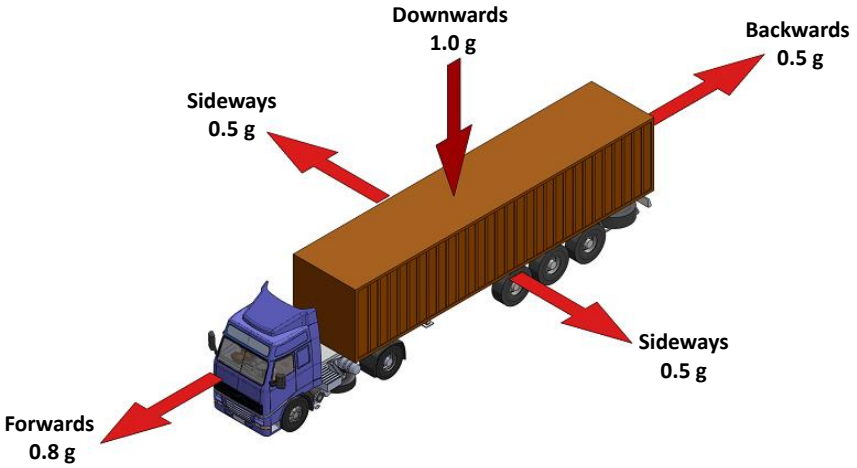
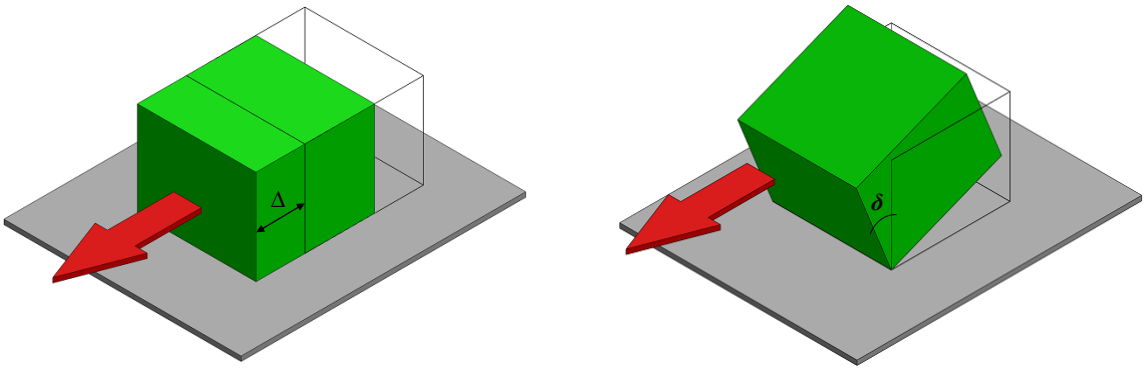


Figure 2.23: Forces acting on the cargo

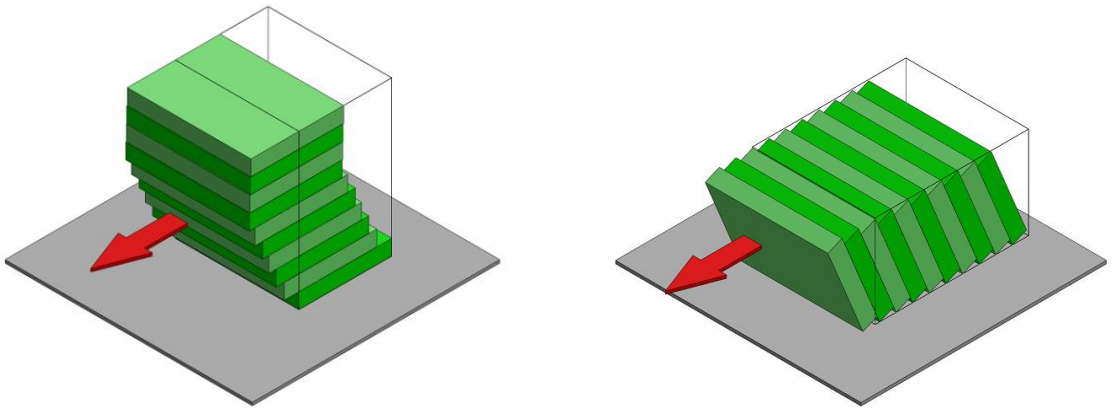
Table 2.6: Reference acceleration values (in g) for different transport modes (IMO/ILO/UNECE, 2014)

Mode of transport		Forwards	Backwards	Sideways	Downwards
Road		0.8	0.5	0.5	1
Railway		0.5	0.5	0.5	1
Sea	Sea area A	0.3		0.5	1
	Sea area B	0.3		0.7	1
	Sea area C	0.4		0.8	1



(a) Rigid sliding (b) Rigid tipping

Figure 2.24: Types of cargo movements



(a) (b)

Figure 2.25: Flexible sliding and tipping

Tables 2.7, 2.8 and 2.9 present results of the computational experiments obtained by authors who have published results for the M1 metric and/or the M2 metric. To the author's knowledge, only five authors have published results for the M1 metric and nine for the M2 metric. All use in the computational tests the 1500 problems proposed by Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999).

The column headings for the computational results in tables 2.7, 2.8 and 2.9 refer to the different algorithms used, namely: H_B_al—an heuristic approach of Bischoff and Ratcliff (1995b); H_BR—an heuristic approach of Bischoff et al. (1995); GA_GB—a genetic algorithm of Gehring and Bortfeldt (1997); H_E—an heuristic approach of Eley (2002); H_B—an heuristic approach of ; GRASP_MO— a GRASP algorithm of Moura and Oliveira (2005); AAR1 and ARR2 — two constructive heuristics of de Araújo and Armentano (2007); TS_L_al—an hybrid tabu search approach of Liu et al. (2011a).

The results for the average percentage of occupied volume of the container by boxes are presented in Table 2.7, for the average M1 metric are presented in Table 2.8 and for the average M2 metric in Table 2.9. The H_E and TS_L_al results were taken from a graphical representation, therefore they may not be entirely accurate.

It can be observed that the AAR2 algorithm holds the best results for the average percentage of occupied volume of the container for BR1 to BR7 problems.

The solutions with the best results for the average M1 for the BR1 to BR7 problems, are from the H_B_al algorithm, while the best results for the average M2 for BR1 to BR7 problems, are from the AAR1 algorithm. For problems BR8 to BR15 only GRASP_MO provides results for M1 and M2.

It can also be observed that the algorithm with the best average percentage of occupied volume does not have the best average values for either M1 or M2. This observation is relevant since it is mentioned in the literature that there is a correlation between stability and a high volume occupation of the container (Bortfeldt and Wäscher, 2013).

Table 2.7: Performance comparison of the percentage of the container volume packed by boxes

Class problem	H_B_al (1995)	H_BR (1995)	GA_GB (1997)	H_E (2002)	H_B (2003)	GRASP_MO (2005)	AAR1 (2007)	AAR2 (2007)	TS_L_al (2011)
BR1	81.76	83.79	85.80	88.05	89.39	89.07	90.86	**91.73**	88.14
BR2	81.70	84.44	87.26	88.44	90.26	90.43	90.88	**91.60**	89.52
BR3	82.98	83.94	88.10	89.23	91.08	90.86	90.94	**91.47**	90.53
BR4	82.60	83.71	88.04	89.24	90.90	90.42	90.67	**91.06**	90.75
BR5	82.76	83.80	87.86	88.99	**91.05**	89.57	90.40	90.90	90.79
BR6	81.50	82.44	87.85	88.91	90.70	89.71	90.14	90.46	**90.74**
BR7	80.51	82.01	87.68	88.36	**90.44**	88.05	89.46	89.54	90.07
BR8	-	-	-	-	-	86.13	-	-	**88.89**
BR9	-	-	-	-	-	85.08	-	-	**88.51**
BR10	-	-	-	-	-	84.21	-	-	**87.76**
BR11	-	-	-	-	-	83.98	-	-	**87.06**
BR12	-	-	-	-	-	83.64	-	-	**86.97**
BR13	-	-	-	-	-	83.54	-	-	**86.90**
BR14	-	-	-	-	-	83.25	-	-	**86.40**
BR15	-	-	-	-	-	83.21	-	-	**86.23**
Mean (BR 1-7)	81.97	83.45	87.51	88.75	90.55	89.73	90.48	**90.97**	90.08
Mean (BR 8-15)	-	-	-	-	-	84.13	-	-	**87.34**
Mean (BR 1-15)	-	-	-	-	-	86.74	-	-	**88.62**

∗ **The best values appear in bold**

Table 2.8: Performance comparison of the M1 metric

M1 Problem	H_B_al (1995)	H_BR (1995)	H_B (2003)	GRASP_MO (2005)	AAR1 (2007)	AAR2 (2007)
BR1	**2.02**	1.13	1.17	1.07	1.15	1.18
BR2	**2.22**	1.10	1.14	1.10	1.15	1.18
BR3	**2.20**	1.08	1.09	1.09	1.12	1.15
BR4	**2.10**	1.07	1.07	1.10	1.11	1.13
BR5	**2.09**	1.06	1.06	1.10	1.11	1.13
BR6	**2.04**	1.06	1.05	1.10	1.10	1.12
BR7	**1.92**	1.04	1.03	1.11	1.08	1.10
BR8	—	—	—	**1.12**	—	—
BR9	—	—	—	**1.10**	—	—
BR10	—	—	—	**1.10**	—	—
BR11	—	—	—	**1.14**	—	—
BR12	—	—	—	**1.15**	—	—
BR13	—	—	—	**1.16**	—	—
BR14	—	—	—	**1.16**	—	—
BR15	—	—	—	**1.17**	—	—
Mean (BR 1-7)	**2.08**	1.08	1.09	1.10	1.12	1.14
Mean (BR 8-15)	-	-	-	**1.14**	-	-
Mean (BR 1-15)	-	-	-	**1.12**	-	-

∗ **The best values appear in bold**

Table 2.9: Performance comparison of the M2 metric

M2 Problem	H_B_al (1995)	H_BR (1995)	GA_GB (1997)	TS_BG (1998)	H_E (2002)	H_B (2003)	GRASP_MO (2005)	AAR1 (2007)	AAR2 (2007)	TS_L_al (2011)
BR1	8.50	10.36	11.00	13.00	9.80	12.37	11.53	**6.00**	10.77	7.40
BR2	11.21	14.60	16.00	19.00	13.50	15.30	12.67	**9.22**	13.72	11.60
BR3	15.93	19.67	18.50	24.50	18.00	17.05	17.75	**10.35**	16.17	15.20
BR4	17.51	23.53	21.50	29.90	20.50	18.65	20.03	**12.48**	18.09	17.80
BR5	21.60	26.03	22.50	34.00	21.50	20.79	22.75	**13.70**	19.51	19.20
BR6	22.13	31.04	25.00	33.50	22.90	23.31	26.50	**15.70**	20.91	22.20
BR7	27.07	35.99	28.50	46.10	26.00	24.25	28.86	**18.24**	23.91	25.20
BR8	—	—	—	—	—	—	**32.77**	—	—	—
BR9	—	—	—	—	—	—	**37.49**	—	—	—
BR10	—	—	—	—	—	—	**39.21**	—	—	—
BR11	—	—	—	—	—	—	**40.63**	—	—	—
BR12	—	—	—	—	—	—	**41.44**	—	—	—
BR13	—	—	—	—	—	—	**41.67**	—	—	—
BR14	—	—	—	—	—	—	**43.14**	—	—	—
BR15	—	—	—	—	—	—	**44.12**	—	—	—
Mean (BR 1-7)	17.71	23.03	20.43	28.57	18.89	18.82	20.01	**12.24**	17.58	16.94
Mean (BR 8-15)	-	-	-	-	-	-	**40.06**	-	-	-
Mean (BR 1-15)	-	-	-	-	-	-	**30.70**	-	-	-

∗ **The best values appear in bold**

2.6 Current status and research focus

There has been significant research on the CLP in the last years. However, the proposed approaches have not been adopted in practice as cargo planning tools by transportation companies. The main reason lies within the fact that there are practical constraints that have a strong influence on cargo arrangements that need to be properly addressed. However, with the number of challenges that the transportation sector faces, transport companies will be compelled to adopt planning cargo tools.

There is, therefore, a pressing need to reduce the gap between research and practical implementation. This reduction can be achieved through the modelling and integration of practical constraints, that, even though they are not something new in the CLP, have been done through an over-simplified representation of reality in some practical constraints.

Cargo stability is one of the most important practical constraints that has had an over-simplified approach. Its importance derives from the fact that it strongly contributes to the safety of cargo transportation, a pressing issue for the transportation industry, and that encompasses a broader set of mechanical properties of boxes that also impact in other practical constraints, that also contribute to the safety of cargo transportation.

The research direction followed in this thesis is the development of an approach to cargo stability that reduces the gap between research and practice.

Chapter 3

Thesis Contribution

The contributions of this thesis can be divided in two parts. In the first part, consisting of Chapter 2, the CLP is presented and framed within the Cutting and Packing problems. The different typologies of CLPs are also presented and a literature review is done focusing on the heuristic approaches. Existing classifications for the heuristic container filling strategies are also presented, as well as a new classification proposal based on the spatial representation and the box filling strategies. A classification of existing papers is also provided. A thorough literature review on the stability constraint is also presented, with a clear distinction between static and dynamic stability approaches. This part of the thesis can be used by researchers, either to get acquainted with the CLP or to have a more comprehensive understanding of the stability constraint. In Chapter 2 we also discuss the need for further research aiming for the inclusion of improved representations of practical constraints in the CLP, to reduce the gap between research and implementation. The second part, which includes the main contribution, consists in new approaches to the stability constraint, grounded on the consideration of two different types of stability, static and dynamic. The research on static stability is presented in the first two scientific papers of Part II of the thesis. The third and fourth scientific papers focus on dynamic stability. A brief introduction to each one of the four papers is presented below, along with some indicators concerning the work dissemination.

3.1 A new approach to static stability

Chapter 5, *A Physical Packing Sequence Algorithm for the Container Loading Problem with static mechanical equilibrium conditions* reports on the development of a feasible loading sequence of boxes, for a CLP solution, while guaranteeing static stability and incorporating operational efficiency constraints.

The paper starts by presenting a literature review covering two main topics within the CLP, static stability and the physical packing sequence. It then introduces a Static Stability Algorithm based on static mechanical equilibrium conditions, developed with the purpose of being used as a stability evaluation function for CLP solutions. The major

benefits of the proposed algorithm include its short processing time, the ability to evaluate static stability each time a box is loaded in the container and the fact that it does not need to take into account the percentage of base support.

It is followed by the presentation of the developed strategy for loading the container. Due to ergonomic factors influencing manual floor loading operations, it was decided to use a loading sequence that would follow a wall building approach, where each box is placed inside the container, starting in its back-bottom-left corner, with boxes placed sequentially along the width of the container and then upwards.

A metric that reflects the effort made by the operator to load a complete cargo arrangement was also introduced, the average *arm's-length* of the loaded boxes. When a box is loaded, the *arm's-length* measures the minor distance from the load position of the box to the first box surface encountered when loading the box from the front of the container.

A Physical Packing Sequence Algorithm is then presented. The intimate relation between the static stability and the loading sequence of boxes inside the container, led to the development of a Physical Packing Sequence Algorithm, that is, an algorithm that, given a container loading arrangement, generates the actual sequence by which each box is placed inside the container, considering static stability and loading operations efficiency constraints.

The computational experiments demonstrate that the proposed static stability approach is less restrictive than the approaches usually found in the literature and is closer to real-world static stability.

The main contributions of this work are the two proposed algorithms, the Static Stability Algorithm based on the static mechanical equilibrium conditions for rigid bodies, and a Physical Packing Sequence Algorithm to evaluate the feasibility of cargo arrangements generated by CLP algorithms. The use of the Static Stability Algorithm embedded in the Physical Packing Sequence Algorithm demonstrated its ability to be combined with other algorithms as an evaluation function.

The results of this research work have been disseminated as follows:

- Presentation of the talk *Analysis of Cargo Stability in Container Transportation* at the IEMS'13 — 4th Industrial Engineering and Management Symposium, held in Porto, Portugal, on the 10^{th} of January, 2013.

- Presentation of the talk *Development of a Cargo Loading Sequence Algorithm for the Container Loading Problem* at the 10th ESICUP Meeting, held in Lille, France, from the 24^{th} to the 26^{th} of April, 2013.

- Presentation of the talk *Desenvolvimento de um algoritmo para a sequência de empacotamento de carga no Problema de Empacotamento em Contentores* at the IO2013, o XVI Congresso da Associação Portuguesa de Investigação Operacional, held in Bragança, Portugal, from the 3^{rd} to the 5^{th} of June, 2013.

- Presentation of the talk *A Physical Packing Sequence Algorithm for the Container Loading Problem with Static Mechanical Equilibrium Conditions* at the XXVI EURO - INFORMS Joint International Conference: "All roads lead to OR", held in Rome, Italy, from the 1^{st} to the 4^{th} of July, 2013.

- Presentation of the talk *Static and dynamic stability constrains within the container loading problem* at the X IWCPRT- 10th International Workshop on Cutting, Packing and Related Topics, held in Gaienhofen-Horn, Lake Constance, Germany, from the 11^{th} to the 15^{th} of September, 2013. The award for the best presentation was given to the author of this thesis.

- This paper has been accepted for publication in the *International Transactions in Operational Research* journal, and is currently in press and available on-line.
  Ramos, A. G., Oliveira, J. F. and Lopes, M. P. (2014), A physical packing sequence algorithm for the container loading problem with static mechanical equilibrium conditions. *International Transactions in Operational Research*. doi: 10.1111/itor.12124.

3.2 Incorporating the static stability algorithm within the CLP

Chapter 6, *A Container Loading Algorithm with Static Mechanical Equilibrium Stability Constraints* reports on the incorporation of the static stability constraint based on the static mechanical equilibrium conditions applied to rigid bodies proposed in Chapter 5, into a constructive heuristic placement procedure, used as part of a multi-population biased random-key genetic algorithm for the SKP and SLOP CLP types of problems.

The paper starts by presenting a literature review covering the CLP and the static stability constraint. It then introduces the proposed CLP algorithm, a multi-population biased random-key genetic algorithm that combines a genetic algorithm, responsible for the evolution of coded solutions (chromosomes), and a constructive heuristic algorithm responsible for decoding the chromosome, generating a solution and evaluating its fitness. This constructive heuristic uses a *maximal-space* representation for the management of empty spaces, and a layer building strategy to fill those *maximal-spaces*. The new stability criterion is used to evaluate stability during the filling of the *maximal-spaces* and the Physical Packing Sequence Algorithm proposed in Chapter 5 is used for the overall evaluation of the loading arrangements.

Three variants of the algorithm are proposed: the first with no stability constraint, the second with the classical full base support constraint and a third one with the new static mechanical equilibrium stability constraint. The proposed algorithms were extensively tested using the well known benchmark instances of Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999) and compared to the best-known results, provided by state-of the-art CLP algorithms found in the literature.

The computational experiments show that, on average, the algorithm variant with the new stability criterion achieves a higher percentage of space utilization than the variant with the classical full base support condition, while fully guaranteeing the static stability of the cargo. Additionally, for problems with strongly heterogeneous cargo (classes BR8 to BR15) the CLP algorithm variant with full base support constraint outperforms the other approaches published in the literature, improving the best solutions known for these classes of problems. The average results for the instances BR-8 to BR15 improve from 91.22% (best known results) to 91.51% using the full support constraint and improved the overall average from 92.87% (best-known results) to 93.92% for statically stable solutions.

The main contributions of this work are the new CLP algorithm for the SKP and SLOP problems with a less restrictive static stability constraint and a CLP algorithm with full base support variant for strongly heterogeneous cargo that outperforms the best algorithms from the literature.

This work has been disseminated as follows:

- Presentation of the talk *Static Stability Algorithm for the Container Loading Problem* at the 11th ESICUP Meeting, held in Beijing, China, from the 19^{th} to the 21^{st} of March, 2014.

- Presentation of the talk *An Algorithm for a Container Loading Problem with Static Mechanical Equilibrium Conditions* at the 20^{th} Conference of the International Federation of Operational Research Societies, held in Barcelona, Spain, from the 13^{th} to the 18^{th} of July, 2014.

- The paper was submitted to the *European Journal of Operational Research* in 2015.

3.3 Developing a physics simulation evaluation tool for dynamic stability within the CLP

Chapter 7, *A physics simulation tool for the container loading problem* reports on the development of a physics simulation tool based on an open-source physics engine, to be used in the simulation of the external loads generated by typical extreme transport vehicle manoeuvres, such as vehicle full braking, tight cornering and fast lane changing.

The paper starts with a brief literature review on physics engines, followed by the presentation of the physics simulation tool, designated StableCargo. The tool is then evaluated regarding its ability to model friction. Friction was considered one of the most important parameters when analysing a physics engine, since the purpose of this physics simulation tool is to simulate the movement of a set of boxes inside a shipping container. A set of benchmark tests was performed, and the results obtained were compared with analytical values, and those obtained using the state-of-the-art engineering simulation software Abaqus Unified FEA.

The results were in agreement with both the analytical values and those obtained using Abaqus FEA, offering good prospects for the use of the tool for evaluating dynamic stability within the CLP.

The main contributions of this work are the physics simulation tool for dynamic stability evaluation of CLP solutions and a set of benchmark tests for physics engines evaluation.

This work has been disseminated as follows:

- Presentation of the talk *A physics simulation tool for the container loading problem* at the 26th European Modeling and Simulation Symposium (EMSS2014), held in Bordeaux, France, from the 10^{th} to the 12^{th} of September, 2014. (presenter: João Jacob)

- A paper has been published in the *Proceedings of the 26th European Modelling and Simulation Symposium (EMSS2014)*, held in Bordeaux, France, from the 10^{th} to the 12^{th} of September, 2014.

- An extended version of the previous paper was submitted to the *International Journal of Simulation and Process Modelling* in 2015.

3.4 Developing of dynamic stability metrics for the CLP

Chapter 8, *Dynamic stability metrics for the container loading problem* reports on the development of new performance indicators for the evaluation of dynamic stability within the CLP, and new dynamic stability metrics for incorporation in container loading algorithms. The paper starts by presenting a literature review covering the dynamic stability constraint in the CLP, and identifies the two most frequently used metrics for dynamic stability evaluation: the mean number of boxes by which items other than those on the floor are supported (M1) and the percentage of boxes with insufficient lateral support (M2). It then presents the two new performance indicators for dynamic stability evaluation, the number of fallen boxes (NFB) and the number of boxes within the Damage Boundary Curve fragility test (NB_DBC). The first one reflects the costly consequences of the fall of cargo during transportation, that is, the damaging of the cargo and the decrease of operations efficiency by the increase of the unloading time or by adding a quality inspection operation. The second performance indicator focuses on the evaluation of the damage that results from mechanical shock, whether as a result of impact between boxes, or between boxes and the container walls. The proposal of the performance indicators is followed by a brief presentation of StableCargo, the physics simulation tool developed in Chapter 7. Due to the computational cost of using StableCargo within a container loading algorithm, two new dynamic stability metrics that model the performance indicators, were proposed. Derived through multiple linear regression analysis, these metrics were found more suitable to incorporate in a container loading algorithm than the existing metrics in the literature.

The main contributions of this work are the new performance indicators and stability metrics for dynamic stability evaluation in the CLP.

This work has been disseminated as follows:

- Presentation of the talk *Dynamic Stability Constraint within the Container Loading Problem* at the IEMS'14 — 5th Industrial Engineering and Management Symposium, held in Porto, Portugal, on the 7^{th} of January, 2014.
- Presentation of the talk *Dynamic stability and product fragility in the CTU loading problem* at the 3rd International Eumos Symposium on cargo securing, transport packaging and safe logistics, held in Brussels, Belgium, from the 6^{th} to the 7^{th} of November, 2014.
- Presentation of the talk *Dynamic stability metrics for the Container Loading Problem* at the 12th ESICUP Meeting, held in Portsmouth, UK, from the 29^{th} to the 31^{th} of March, 2015.
- The paper was submitted to *Transportation Research Part C* in 2015.

Chapter 4

Conclusion

In this chapter, the main achievements and contributions of the thesis are reviewed. It starts with the summary of the thesis, followed in Section 4.2, by a short list of the major contributions. In Section 4.3, possible directions for future research within the CLP are discussed.

4.1 Discussion

The main aim of this PhD project was to develop a new approach to the stability constraint in the CLP under a realistic framework. That was accomplished through a literature research presented in Chapter 2 and four research projects, each one described in detail in Chapters 5-8.

Despite the increasing attention that CLP has received in the literature in the last years, there is still a gap in the adoption of the proposed solutions by the transport industry. This is mainly due to the existence, in real-word problems, of practical-relevant constraints that have a strong influence in cargo arrangements. Among these practical-relevant constraints, cargo stability is considered to be one of the most important ones. However, it as been addressed under over-simplified assumptions, not responding to the needs in practice. Therefore, in this thesis an approach to stability was developed within a realistic framework, with the assumption that the study of the static stability of cargo, that is, the stability during loading operations, must be separated from the study of dynamic stability, that is, the stability of cargo during transportation. In the thesis, Chapters 5 and 6 address the static stability of cargo and Chapters 7 and 8 address the dynamic stability of cargo.

In Chapter 5 two algorithms are proposed. The first is the Static Stability Algorithm that evaluates the static stability of a cargo arrangement based on static mechanical equilibrium conditions applied to rigid bodies derived from Newton's laws of motion. The second algorithm builds on the first algorithm to guarantee static stability, and generates the actual sequence by which each box is placed inside the container, given a cargo loading arrangement. A metric to evaluate the effort made by the operator during the load of a

complete cargo arrangement, designated *arm's-length*, is also introduced.

The proposed approach to static stability has proven to be less restrictive than the classical full base support approach and is closer to real-world static stability. It was also demonstrated that the static stability algorithm can be combined with other algorithms as an evaluation function.

In Chapter 6 the work on static stability is continued and a new container loading algorithm proposed. The algorithm, a multi-population biased random-key genetic algorithm, combines a genetic algorithm with a constructive heuristic algorithm responsible for decoding the chromosome, generating a solution and evaluating its fitness. The constructive heuristic has three variants that use a *maximal-space* representation for the management of empty spaces, and a layer building strategy to fill those *maximal-spaces*. One variant does not enforce any static stability constraint, another enforces full base support and another incorporates the new static mechanical equilibrium stability constraint and the physical packing sequence algorithm presented in Chapter 5. With the new stability criterion it was possible to achieve, for each class of problem instances, on average, a higher percentage of space utilization compared with the classical full base support condition, while at the same time guaranteeing the static stability of the cargo. The proposed algorithm with the full base support variant improved, on average, the best-known results for the more heterogeneous classes of problems.

The approach to dynamic stability was first addressed in Chapter 7. It presents and tests StableCargo, a physics simulation tool based on the physics engine Bullet, developed for the evaluation of cargo loading arrangements during transportation.

This simulation tool is used in Chapter 8 to determine the value of the two new performance indicators developed for the measurement of the dynamic stability of a cargo arrangement, the number of fallen boxes and the number of boxes within the Damage Boundary Curve test. Due to the computation effort required by StableCargo and the subsequent difficulty of integrating it in a container loading algorithm, two new dynamic stability metrics, computed by multiple linear regression, that model the proposed stability performance indicators are presented. The computational results show that the proposed metrics are a better surrogate measure than the dynamic stability metrics currently used in the literature.

4.2 Main contributions

The main contributions of this thesis consist in the refinement of the stability constraint in the CLP. Stability was explored separating static from dynamic stability, and through this work a new approach was developed both for static and dynamic stability. This thesis makes the following contributions to CLP with stability constraints:

- a Static Stability Algorithm based on the static mechanical equilibrium conditions for rigid bodies;

- a Physical Packing Sequence Algorithm to evaluate the feasibility of cargo arrangements generated by CLP algorithms;
- a metric that reflects the effort made by the operator to load a complete cargo arrangement - the *arm's-length* of a loaded box;
- a new procedure for filling the *maximal-spaces* that allows the evaluation of static stability when there are gaps between supporting boxes;
- a new hybrid genetic algorithm based on a multi-population biased random key genetic algorithm and a constructive heuristic for the CLP, with three variants, one with no stability constraint, another with the full base support constraint and another with static mechanical stability conditions;
- a container loading algorithm for the full base support variant of the CLP that improved the best known average results for the instances BR-8 to BR15.
- a container loading algorithm that improved the best known average results for the instances BR-1 to BR15, incorporating static stability constraints.
- the development of StableCargo, a physics simulation tool to evaluate dynamic stability within the CLP.
- two new performance indicators to evaluate dynamic stability within the CLP;
- two dynamic stability metrics derived from the dynamic stability performance indicators to be incorporated in a container loading algorithm.

4.3 Future Work

Clearly, one of the main directions for future research in the CLP lies within the development of new approaches to the practical-relevant CLP constraints, to continue the approach movement between research and practice.

In particular, three interesting research topics deserve great attention in future research in CLP:

- Incorporate safety in the CLP. Safety in its true meaning has not been studied in the CLP. The concept of safety in CLP is mainly associated with stability, due to its relevance and the research effort spent on it. However, safety is not only derived from stability. If the weight limit constraint is not ensured or the weight distribution of the boxes in the CTU is not taken into account, accidents may occur. Also, if the orientation of the box and its load bearing strength is not guaranteed, safety is not assured. The research on safety has to be developed as the integration of several practical constraints which include stability, weight limit, weight distribution, orientation, stacking and positioning. Safety can be addressed not only as a set of

constraints but also as a goal. In this case, it is also required to develop multi-objective approaches for the CLP.

- New CLP instances. One of the reasons that may contribute to the discrepancy between research and practice is the absence of problem instances, for research purposes, that include a large spectrum of practical relevant constraints. It is therefore a pressing need to develop new test instances that fully reflect real world constraints, and therefore promote the development and benchmark of new algorithms that can be effectively used in practice.

- New taxonomy for the CLP. Closely related to the previous point, are the limitations of the existing taxonomy for the C&P problems of Wäscher et al. (2007) and the classification structure proposed by Bortfeldt and Wäscher (2013). Even though they help to structure the way the CLP and its practical constraints have been tackled in the literature, at the same time they perpetuate the inadequacy of how the CLP and its practical-relevant constraints have been addressed. A new taxonomy for the CLP that is suitable for the existing solutions and considers present and future needs of the transportation industry, would provide an extremely important tool for researchers and would contribute for the development of the field and to diminish the gap between research and practice.

Part II

Scientific Papers

Chapter 5

A Physical Packing Sequence Algorithm for the Container Loading Problem with static mechanical equilibrium conditions

A. Galrão Ramos[†] [‡], José F. Oliveira[†] and Manuel P. Lopes[‡]

[†] INESC TEC and Faculty of Engineering, University of Porto
[‡] CIDEM, School of Engineering, Polytechnic of Porto

Abstract

The Container Loading Problem (CLP) is a combinatorial optimization problem for the spatial arrangement of cargo inside containers so as to maximize the usage of space. The algorithms for this problem are of limited practical applicability if real-world constraints are not considered, one of the most important of which is deemed to be stability. This paper addresses static stability, as opposed to dynamic stability, looking at the stability of the cargo during container loading. This article proposes two algorithms. The first is a Static Stability Algorithm based on static mechanical equilibrium conditions that can be used as a stability evaluation function embedded in CLP algorithms (e.g. constructive heuristics, metaheuristics). The second proposed algorithm is a Physical Packing Sequence Algorithm that, given a container loading arrangement, generates the actual sequence by which each box is placed inside the container, considering static stability and loading operations efficiency constraints.

5.1 Introduction

The transportation sector is currently facing a number of challenges within Europe. The European Commission has cited an efficient transport system as essential to maintaining the EU's prosperity; faced with the present scenario of rising oil prices, growing congestion and looming climate change, a new strategy for the transport systems in Europe is needed. The strategy established by the EU aims to achieve a 60% reduction in CO_2 emissions and oil dependency by 2050, without limiting freedom of movement. A strong focus is laid on optimizing the performance of multimodal logistic chains, and on increasing the efficiency of transport with information systems (EC, 2011).

It is thus expected that in the next decades freight transportation costs will rise, and the need for increased efficiency and security in the transportation sector will drive companies to adopt cargo planning tools that can be effectively used in practice.

It is not then surprising the amount of research that has been carried out focusing on the optimization of the spatial arrangement of cargo inside transportation vehicles or containers.

This problem is known in the literature as the Container Loading Problem (CLP), which belongs to the more generic combinatorial optimization class of Cutting and Packing problems. In the Container Loading Problem, a set of rectangular shaped boxes (small items) must be packed orthogonally in a set of containers (three dimensional, rectangular, large objects), in a way that the boxes do not overlap and all the boxes of the subset lie entirely within the container. As an assignment problem there are two possible objectives: output value maximization and input value minimization. The former refers to problems where the number of containers is not sufficient to accommodate all the boxes and therefore the objective is to maximize the value of boxes assigned to all the containers. The latter refers to problems where the number of containers is sufficient to accommodate all the boxes, such that the objective is to minimize the cost of the containers used (Wäscher et al., 2007).

However, the majority of the approaches to the CLP found in the literature are of limited applicability in practical situations. The existence of a series of real-world constraints that strongly influence the loading pattern of a container contributes to that limitation (Bortfeldt and Gehring, 2001; Bischoff and Ratcliff, 1995a). One of these constraints is cargo stability, which several authors consider to be the most important.

This paper aims to help reduce the discrepancies between scientific outcomes and real-world needs, firstly, by refining the concept of cargo stability and developing a new static stability evaluation function within the CLP and, secondly, by determining the actual loading sequence of a given packing arrangement, taking into account static stability and loading operations efficiency constraints. The approach followed is based on the assumption that stability should be addressed by separating static stability from dynamic stability, i.e., stability of the cargo during loading operations from stability during transportation.

This paper presents a Static Stability Algorithm (SSA), that approaches static stability based on the static mechanical equilibrium conditions applied to rigid bodies derived from Newton's laws of motion, and a Physical Packing Sequence Algorithm (PPSA), that

evaluates the feasibility of an existing loading arrangement that uses the Static Stability Algorithm to evaluate the box placement stability.

The remainder of this paper is organized as follows. Section 5.2 presents a literature review covering the two main topics of this paper, static stability and the physical packing sequence. In Section 5.3, an approach to static stability is presented and an algorithm for static stability evaluation is proposed. It is followed in Section 5.4 by the presentation of the efficiency approach to the loading operations. In Section 5.5, a physical packing sequence algorithm is proposed, followed by the presentation of computational results in Section 5.6. Finally, Section 5.7 draws some conclusions from the findings.

5.2 Literature review

5.2.1 Static stability

The stability constraint is considered one of the most important Container Loading Problem constraints. In a recent literature review of container loading constraints, Bortfeldt and Wäscher (2013) note the particular relevance of stability constraints in the literature. Previous works that address cargo stability use the term "stability" in a rather simplified way, and often as if the term was self-explanatory. Different concepts, such as loading operations stability and transportation stability, are sometimes not addressed separately but dealt with indiscriminately. In the literature, some approaches to stability make a distinction between static (vertical) and dynamic (horizontal) stability, that is, the stability of cargo during loading operations and the stability of cargo during transportation, but the majority of literature approaches only focus on static stability (Bortfeldt and Wäscher, 2013).

Three different types of approaches to static stability can be found in the literature: the full base support, the partial base support and the static mechanical equilibrium approach. Of the three approaches, full base support guarantees static stability, while the proposed approaches for partial base support and static mechanical equilibrium do not guarantee static stability.

The full base support approach requires the entire base of a box to be in contact with the base of the container or with the top surface of other boxes. As a result no overhanging boxes are allowed. Examples can be found in Bischoff and Ratcliff (1995a), Bortfeldt and Gehring (2001), Gonçalves and Resende (2012) and Zhu and Lim (2012). This approach excessively penalizes space usage in the container (Bischoff and Ratcliff, 1995a), and does not necessarily meet real-world needs.

In the case of partial base support, either the entire base of a box is required to be in contact with the base of the container, or a pre-specified percentage of the area of a box base must be in contact with the top surface of other boxes. As an example, Carpenter and Dowsland (1985) require the contact area to fall in the range 95% to 75%, while Christensen and Rousøe (2009) require a minimum of 80%, Gendreau et al. (2006), Fuellerer et al. (2010) and Tarantilis et al. (2009), 75%, Gehring and Bortfeldt (1997), 70%

and Mack et al. (2004), 55%.

In the static mechanical equilibrium approach it is required that:

- the entire base of a box be in contact with the base of the container or,
- the centre of gravity of a box be located above the contact surface of the supporting boxes.

Even though the centre of gravity condition is not referred to in the literature as a static mechanical equilibrium condition (Mack et al., 2004; Lin et al., 2006) it does in fact derive from static mechanical equilibrium conditions applied to rigid bodies. The first authors to actually mention the conditions within the CLP are de Castro Silva et al. (2003). Their approach to static stability is divided into two parts. In the first part it is checked if at least one of the following three conditions is met: (1) the base of the box is fully in contact with the base of the container; (2) the centre of gravity of the box is directly above a supporting box; (3) the centre of gravity of the box is not directly above a supporting box, but there is a vertical plane passing through the centre of gravity and connecting two non-adjacent supporting boxes. The second part addresses the resultant moments for a set of boxes, stating that if the sum of the moments for every box of the set is null then the set is considered to be stable. The approach presented by de Castro Silva et al. (2003) has two practical limitations: no algorithm or method is proposed for checking condition 3 (the existence of a vertical plane) or for evaluating the sum of the moments.

Another aspect of recent approaches in the literature is the replacement of static stability constraints by the full base support constraint (Gonçalves and Resende, 2012; Zhu and Lim, 2012). This is particularly clear in the performance evaluation of the proposed solutions. The metrics used for performance benchmarking algorithms are usually the percentage of volume loaded with or without full base support, and consequently the solutions are obtained by imposing full base support, thus assuring stability. As a result, the goal is not to obtain a loading arrangement that is statically stable but a loading arrangement where all boxes have full base support.

5.2.2 Physical packing sequence

Evaluating the stability of a cargo arrangement during loading operations is intimately related with the actual packing sequence of boxes inside the container. The physical packing sequence is the sequence by which each box is placed inside the container in a specific location determined by the CLP algorithm. This sequence is unlikely to correspond to the generation order of the positions and spatial orientation for the boxes in the cargo arrangement (Ngoi et al., 1994; Bischoff and Ratcliff, 1995a).

The generation of a cargo loading sequence to be used in practice is not taken into account by the majority of existing CLP algorithms which focus on space optimization. This issue is only tackled in situations where the loading or unloading sequence is directly related with the problem characteristics (e.g. multi-drop situations), or is directly related

with the nature of the problem, as when the CLP is combined with the Vehicle Routing Problem (VRP) (Bortfeldt and Wäscher, 2013).

Even though the physical packing sequence is usually ignored in the existing CLP algorithms, some authors mention the possibility of using an algorithm which is independent of that in the CLP to generate the packing sequence. However, no such algorithm is typically provided (Ngoi et al., 1994; Bischoff and Ratcliff, 1995a; Bortfeldt et al., 2003).

The physical packing sequence is also dependent on the presence of loading or unloading constraints and is usually addressed together with the ease of the loading operation. The ease of the loading operation reflects the effort required to handle the boxes during loading.

The ease of loading or unloading is also usually addressed in problems where the CLP and the VRP are combined (Gendreau et al., 2006; Moura and Oliveira, 2008; Tarantilis et al., 2009; Fuellerer et al., 2010; Iori and Martello, 2010) or in situations of cargo multi-dropping (Junqueira et al., 2012a, 2013; Ceschia and Schaerf, 2011; Christensen and Rousøe, 2009). In both situations, boxes have to be delivered to different destinations (customers) and must be located close to each other inside the container. The goal is to have all the boxes of a destination unloaded without moving any of the boxes of each of the subsequent destinations. This would make unloading easier and avoid unnecessary unloading and reloading operations (Moura and Oliveira, 2008; Fuellerer et al., 2010).

The loading arrangement, when different destinations are considered, must take into account the sequence by which each customer is served and, therefore, the sequence by which the boxes are unloaded. Usually a *Last in First Out* (LIFO) strategy is followed so that boxes can be unloaded without the need to rearrange other boxes. This means that in order to unload a box, no other box can be placed either in front or above it. This geometric condition is treated in the CLP as a hard constraint (Christensen and Rousøe, 2009).

Besides the LIFO constraint, other constraints are also considered by some authors when addressing the feasibility of solutions to the CLP under multi-drop situations. The maximum reach is a parameter used to show the maximum number of units of length by which the worker can exceed the previous customer's virtual plane, in order to arrange the boxes of the next customer. The virtual plane of a customer is a plane parallel to the front of the container, defined by the customer box surface closest to the container entrance. This parameter, δ_k, illustrated in Figure 5.1, can also represent the reach of the worker's arm or even a device used to load/unload the boxes (Junqueira et al., 2012a, 2013).

In the context of the home delivery model, which can be considered an extreme case of the multi-drop situation, Liu et al. (2011b), discuss the unloading cost. The unloading cost measures the effort involved in unloading a box from a container. The authors define the unloading cost of a box as being directly proportional to the number of boxes that have not yet been delivered and that need to be unloaded and reloaded in order to unload the box.

To calculate the unloading cost Liu et al. (2011b) use the invisible and untouchable rule. According to this rule, a box that has not been delivered, counts as a cost to another box if it is positioned in front of the box or makes the box untouchable by the worker. To evaluate the latter condition two variables were defined, touchable length $L_{touchable}$ and

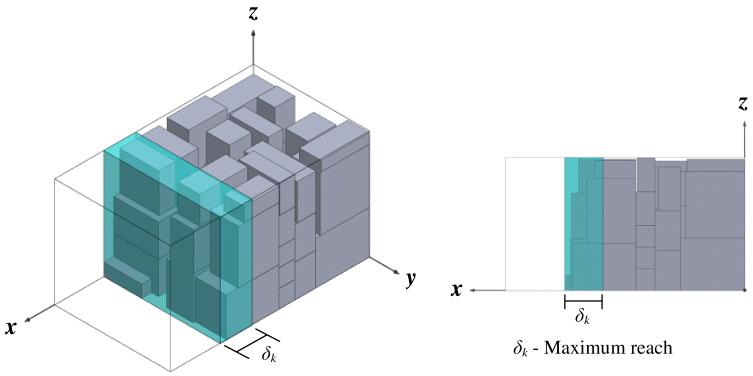


Figure 5.1: Maximum reach parameter

touchable height $H_{touchable}$, which are the distances from the box to the feet of the worker along the x and z-axis respectively (see Figure 5.2).

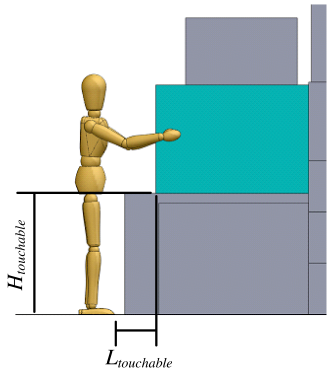


Figure 5.2: Untouchable box parameters

According to these authors, if a box can be touched by the worker the variables should satisfy two constraints:

$$L_{touchable} + H_{touchable} \leq (body\ height + arm\ length)$$

$$L_{touchable} \leq min\{(body\ height + arm\ length - z_i), arm\ length\}$$

where z_i is the z-axis coordinate origin of box b_i.

5.3 Static stability

Different strategies have been presented in the literature to address the static stability constraint. The majority of authors require either full base support or almost full base support for the boxes as a condition to guarantee static stability. Considering that these

conditions can be very restrictive during the loading of boxes into the container, an approach to stability is presented here based on the static mechanical equilibrium conditions applied to rigid bodies.

There are two conditions of equilibrium for rigid bodies that derive from Newton's first and third laws of motion. The first law states that "Every object persists in its state of rest or uniform motion in a straight line unless it is compelled to change that state by forces impressed on it". The third law states that "For every action there is an equal and opposite reaction". In other words, if object A exerts a force on object B, then object B also exerts an equal and opposite force on object A (Hibbeler, 2010b).

The first condition of equilibrium is

$$\sum \overrightarrow{F} = \overrightarrow{0} \tag{5.1}$$

where $\overrightarrow{F}$ represents the external forces applied on a rigid body. A force is a vector quantity that characterizes the force magnitude and the direction of its action (Hibbeler, 2010b).

The second condition of equilibrium is

$$\sum \overrightarrow{M_O} = \sum (\overrightarrow{r} \times \overrightarrow{F}) = 0 \tag{5.2}$$

where $\overrightarrow{r}$ represents the vector from point O to the line of action of force $\overrightarrow{F}$. $\overrightarrow{M_O}$ represents the moment of a force, i.e., the tendency of a force to rotate a body about a point or axis. The first condition provides the translational equilibrium and the second condition the rotational equilibrium of a body (Hibbeler, 2010b).

Following Newton's Laws the mass of a system can be treated as a point mass. In the proposed approach boxes are considered bodies of homogeneously distributed mass. When the body is homogeneous the centre of gravity of the body coincides with the geometric centre of the body. The centre of gravity is the point where an object's weight distribution is considered to be applied (Hibbeler, 2010b).

When evaluating the static stability of a cargo arrangement we have to deal with a box arrangement of several items that can move independently of each other, that is, there are several rigid bodies which all have to be in a stable position.

For each box we have the static stability criteria (5.1) and (5.2).

Let b be a box placed at position (x, y, z) with depth (d) width (w) and height (h), whose support polygon is denoted by S. The support polygon of a box is formed by the convex hull of all points in contact with the base of the box. The support polygon concept is frequently used in the research field of human movement simulation for stability modeling purposes (Badler et al., 1980; Vukobratović and Borovac, 2004).

Given that the only external force acting on the set of boxes is the gravitational force, and all the faces of the boxes are parallel or perpendicular to the direction of the gravitational force, when taken in conjunction with (5.1) this implies that all the forces acting on each of the boxes are parallel to the gravitational force.

Box b is subject to the gravitational force $(\overrightarrow{W})$, the action forces $(\overrightarrow{A_k})$ of supported boxes k and the reaction forces $(\overrightarrow{R_j})$ of support boxes j. The reaction forces are acting on

the area of the base of the box in contact with other boxes. Let $\overrightarrow{A} = \sum_{k=1}^{K} \overrightarrow{A_k}$ applied at point M so that $\overrightarrow{A} \times \overrightarrow{OM} = \sum_{k=1}^{K} (\overrightarrow{A_k} \times \overrightarrow{OM_k})$ and $\overrightarrow{R} = \sum_{j=1}^{J} \overrightarrow{R_j}$ applied at point Q so that $\overrightarrow{R} \times \overrightarrow{OQ} = \sum_{j=1}^{J} (\overrightarrow{R_j} \times \overrightarrow{OQ_j})$. Let point O be the origin of the coordinate system. Figure 5.3 illustrates three stacked boxes A, B and C, and the free body diagram of box B, which depicts all the relevant forces acting on the box.

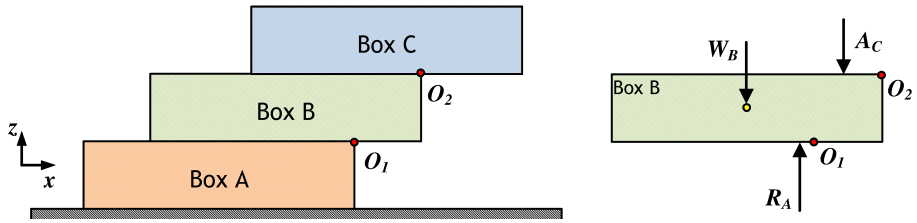


Figure 5.3: Free body diagram of box B

Let $\overrightarrow{F_R}$, be the resultant force acting downwards parallel to the z-axis on box b, acting on point P with coordinates (x_p, y_p, z_p). The magnitude of the resultant force $\overrightarrow{F_R}$ is established by adding $\overrightarrow{W}$ and $\overrightarrow{A}$ i.e.,

$$\overrightarrow{F_R} = \overrightarrow{W} + \overrightarrow{A}$$

Application point P is determined by calculating the moment of $\overrightarrow{F_R}$ about point O, and the resulting moment of $\overrightarrow{W}$ and $\overrightarrow{A}$. The moment of $\overrightarrow{F_R}$ about point O equals the sum of moments of the forces $\overrightarrow{W}$ and $\overrightarrow{A}$ about O, that is,

$$\overrightarrow{M_{F_R}} = \overrightarrow{M_W} + \overrightarrow{M_A}$$

$$\overrightarrow{F_R} \times \overrightarrow{OP} = \overrightarrow{W} \times \overrightarrow{OG} + \overrightarrow{A} \times \overrightarrow{OM}$$

where $\overrightarrow{OP}$, $\overrightarrow{OG}$ and $\overrightarrow{OM}$ are radius vectors from the origin of the coordinate system O to the point where the resultant force acts (P), centre of gravity of box b (G), and the points where the action forces are applied (M), respectively.

Proposition 1. *Stability criteria (5.1) and (5.2) are met for a box b, subject to the force* $\overrightarrow{F_R}$*, if the projection of the application point P with coordinates* (x_p, y_p, z_p) *in plane* $(0, 0, z_b)$*, lies inside the support polygon S of box b.*

Proof. Applying equations (5.1) and (5.2)

$$\sum \overrightarrow{F} = \overrightarrow{F_R} + \overrightarrow{R} = \overrightarrow{0}$$

$$\sum \overrightarrow{M_O} = \overrightarrow{F_R} \times \overrightarrow{OP} + \overrightarrow{R} \times \overrightarrow{OQ} = \overrightarrow{0}$$

reveals that coordinates (x_p, y_p) are the same as coordinates (x_q, y_q) of point Q.

Since the application point of the resultant force of two parallel forces with the same direction and with application points a and b in the same plane perpendicular to the forces,

is located in a line segment that connects a and b such that $Q = \lambda a + (1-\lambda)b$ for $0 \leq \lambda \leq 1$, the resultant of the reaction forces acting on the base of a box is always located inside a convex hull defined by all the horizontal support points of the box.

It can then be concluded that if P is inside the support polygon S, then conditions (5.1) and (5.2) are satisfied. □

5.3.1 Static Stability Algorithm

The exact task of the Static Stability Algorithm is as follows: given a stable arrangement A' of n boxes, the SSA has to decide whether the extended arrangement A consisting of A' and one additional placement (b, x, y, z, d, w, h) is also stable. Hence, we have to answer the question of which placements of A might become unstable by the new placement of b. If the set of these placements $A_r \subseteq A$ is determined, then the task of the SSA is to perform a stability check for the new placement of b and for all placements in A_r.

A placement of a box b consists of a reference point with coordinates (x, y, z) and depth (d), width (w) and height (h), dimensions measured along the x, y and z-axis of a Cartesian coordinates system, after being positioned in the container. The container is considered to be placed in the first octant of the coordinates system with the back-bottom-left corner lying in its origin.

Proposition 2. *The set of possibly affected placements A_r consists of all placements of boxes that support box b and support the supporting boxes of b etc.*

Proof. The proof is clear, since the weight transmission can only take place from b to the boxes in A_r. □

In the proposed Static Stability Algorithm, a box b located at (x_b, y_b, z_b) subject to the force F_R, is considered stable, if the projection of the application point P with coordinates (x_p, y_p, z_p) in plane $(0, 0, z_b)$, lies inside the box b support polygon S. This condition requires that each time an attempt is made to place a box inside the container, the static stability of the subset of boxes A_r loaded in the container must be evaluated, in order to check if the new subset A is stable. Therefore, to guarantee that the static stability condition holds, for all boxes of A_r, one of the three following conditions must be satisfied:

1. the support polygon of box b is the container floor, i.e., whenever z_b, the z-axis coordinate of a box, equals zero;
$$z_b = 0$$
2. point P lies in the interior or frontier of a support polygon, defined by the top edges of a support box b_j (Figure 5.4). The condition holds for box b, supported by box b_j located at (x_j, y_j, z_j) and with dimensions (d_j, w_j, h_j), when
$$x_j \leq x_p \leq x_j + d_j$$
$$y_j \leq y_p \leq y_j + w_j$$
$$z_b = z_j + h_j$$

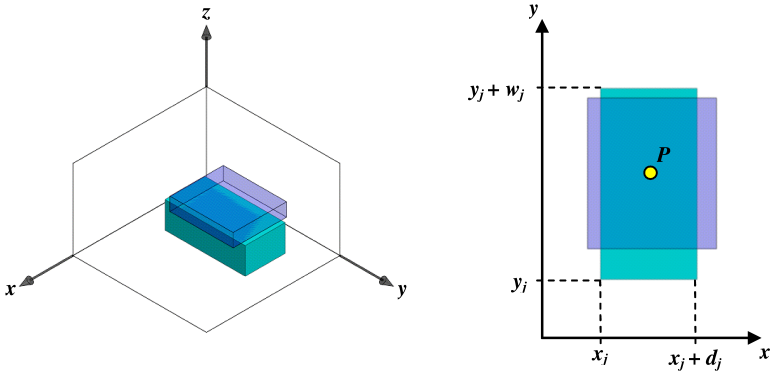


Figure 5.4: Example support polygon for condition 2

3. point P lies in the interior or frontier of a support polygon S, a convex polygon defined by the convex hull of the vertices of the polygons generated by the intersection of box b with its supporting boxes (Figure 5.5). Consider S to be a convex polygon in plane $(0, 0, z_b)$ with m vertices $(p_0, p_1, ..., p_{m-1})$ defined by a sequence of points $(x_0, y_0), (x_1, y_1), ..., (x_{m-1}, y_{m-1})$ given in counterclockwise direction. Condition 3 holds whenever point P is an interior point of support polygon S. Point P, is a point or an interior point of support polygon S, if P is located on a line segment that connects a pair of points a, b in S, such that $P = \lambda a + (1 - \lambda)b$ for $0 \leq \lambda \leq 1$ (LaValle, 2006).

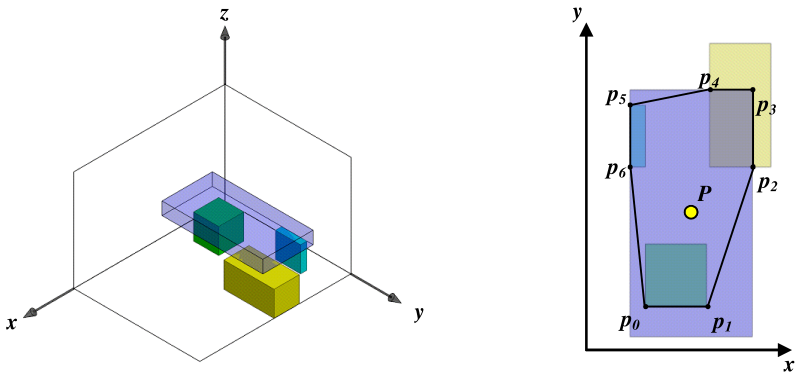


Figure 5.5: Example support polygon for condition 3

The Static Stability Algorithm is described in Algorithm 1.

Construction of the support polygon

In order to be able to evaluate stability condition 3, it is necessary to build the convex polygon induced by the support boxes. The support polygon S represented by vertices $(p_0, p_1, ..., p_{m-1})$ is the convex hull of the set of points T, which represents the vertices

Algorithm 1 Static_Stability (box b)

Input: Let box b be a box placed at coordinates (x_b, y_b, z_b) with depth (d_b) width (w_b) and height (h_b).

Output: Let *stable* be a boolean variable where TRUE represents a stable box and FALSE an unstable box.

Begin
stable ← FALSE
if $z_b = 0$ **then**
    stable ← TRUE
    return *stable*
end if
Let U be the set of loaded boxes b_j that support box b
Let F_r be the sum of downward forces acting on box b
Let P be the acting point of F_r with coordinates x_p, y_p
for each box $b_j \in U$ **do**
    if $(x_j \leq x_p \leq x_j + d_j)$ and $(y_j \leq y_p \leq y_j + w_j)$ **then**
        stable ← TRUE
    end if
    Determine the intersection vertices v with box b
end for
if *stable* = FALSE **then**
    $S \leftarrow$ **Call** Gift wrapping (v) ▷ Determine box b support polygon
    stable ← **Call** Point-in-Polygon (P, S) ▷ Determine box stability
end if
if *stable* = TRUE **then**
    Determine box b reaction forces R_j
    for each box $b_j \in U$ **do** ▷ Determine system stability
        if Call Static_Stability(b_j) = FALSE **then**
            stable=FALSE
        end if
    end for
end if
return *stable*
End

$v_{j,k} = (x_{j,k}, y_{j,k})$ for $k = (1, 2, .., 4)$ of the intersection of box b with its support boxes b_j for $j = (1, 2, ..., n)$. The intersection of a box b with its support box b_j has the shape of a rectangle, since boxes are placed orthogonally to each other.

There are four types of intersection of two dimensional orthogonal objects. Let a and b be two orthogonal rectangles. Rectangle a intercepts b if at least one of the following conditions, illustrated in Figure 5.6, stand (Edelsbrunner and Maurer, 1981):

1. a contains the left bottom point of b;
2. the left border line of a intercepts the bottom border line of b;
3. the bottom border line of a intercepts the left border line of b;
4. the left bottom point of a is contained in b.

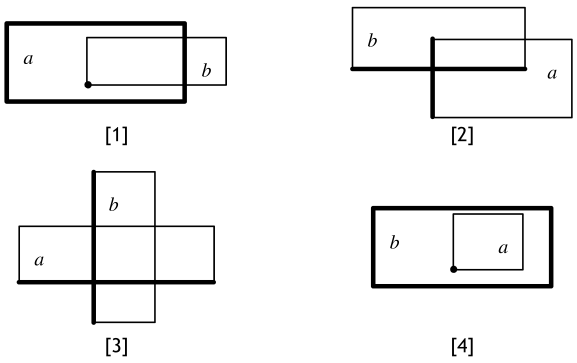


Figure 5.6: The four possible types of intersection (Adapted from Edelsbrunner and Maurer (1981))

Based on the four types of intersection and considering two orthogonal rectangles b and b_j, each defined by the lower left and upper right corner with coordinates $(x_b, y_b), (x_b + d_b, y_b + w_b))$ and $((x_j, y_j), (x_j + d_j, y_j + w_j))$, the vertices of the intersection area can be determined by the following expressions:

$$v_{j,1} = (max(x_b, x_j), max(y_b, y_j))$$

$$v_{j,2} = (min(x_b + d_b, x_j + d_j), max(y_b, y_j))$$

$$v_{j,3} = (max(x_b, x_j), min(y_b + w_b, y_j + w_j))$$

$$v_{j,4} = (min(x_b + d_b, x_j + d_j), min(y_b + w_b, y_j + w_j))$$

The algorithm that generates the convex hull of T is based on the concept of *Gift Wrapping* which basically simulates the wrapping of a taut piece of paper around a gift (Chand and Kapur, 1970; Jarvis, 1973; Dibakar and Mruthyunjaya, 1999).

It starts by computing the point p_0 with the lowest $y_{j,k}$, since it is known that the most inferior point must be a convex hull vertex. The algorithm proceeds, finding the next convex hull vertex with the smallest counterclockwise angle which is greater than zero with respect to p_0. Once the successive point in the convex hull is identified, the process continues based on angles with respect to the new point until the next vertex is the initial point. The algorithm also eliminates collinear points. Figure 5.7 illustrates the algorithm.

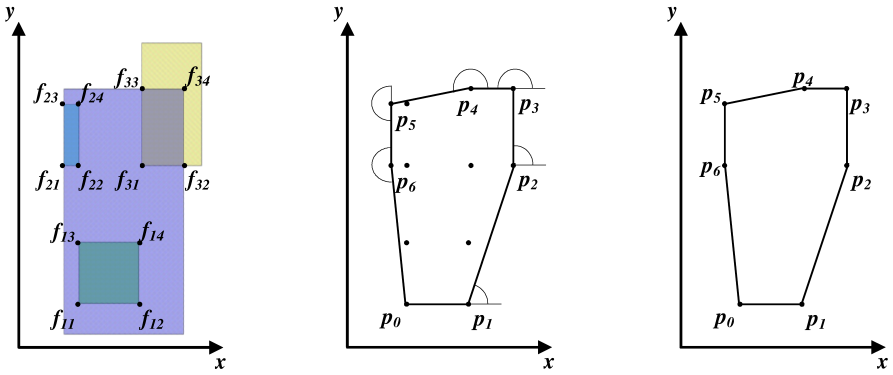


Figure 5.7: *Gift Wrapping* algorithm illustration

The pseudo code of the *Gift Wrapping* algorithm is described in Algorithm 2. The function *findangle* in the algorithm returns the angle between the horizontal line and the line joining two points.

Determine if the application point is on the boundary or inside the support polygon

Following the construction of the support polygon it is necessary to verify if the *application point* is placed on the boundary or inside the polygon. Testing whether a point is inside or on the boundary of a polygon is known in computational geometry as the point-in-polygon problem. The strategy followed to solve the problem is to test the point against each exterior edge in sequence. If the point is outside any edge, then the point must be outside the convex polygon (Shimrat, 1962; Haines, 1994).

The algorithm uses the properties of the vector product between two vectors in order to establish if the point is to the left, to the right or on the edge. It starts by generating two vectors. One is from the first point (p_i) to the final point (p_{i+1}) of the segment, and the other is from the first point (p_i) of the segment to the application point (p_P). It then calculates the result of the vector product for the vectors generated, and conclusions are drawn from there, as shown in Figure 5.8. In this case, $\overrightarrow{p_i p_{i+1}} \times \overrightarrow{p_i p_P}$ is positive if p_P is to the right of the segment, it is negative if p_P is to the left or it is 0 if it is on the segment. Since the polygon was constructed in a counterclockwise direction, it is considered that p_P is inside the polygon if the vector product is negative for all segments of the polygon. The pseudo code of the Point-in-Polygon Algorithm is described in Algorithm 3. The algorithm is an adaptation of the one proposed by Nordbeck and Rystedt (1967).

Algorithm 2 *Gift Wrapping* (points)

Input: Let [points] be a vector of $NumPoints$ points with coordinates x_i, y_i, $i = 1, ..., NumPoints$. Let $used_i$ be a flag that indicates if a point was added to the convex hull.

Output: Let S be the convex hull of [points].

Begin
for $j = 1$ **to** $NumPoints$ **do**
  $[points][j].used \leftarrow$ FALSE
end for
Let $minPoint$ be the point with the minimum y-axis coordinate (if there is more than one $minpoint$, select the one with the minimum x-axis coordinate)
Call addConvexHullPoint($minPoint$) ▷ add point to convex hull S
$points[minPoint].used \leftarrow$ TRUE
$currPoint \leftarrow minPoint$ ▷ set the current point
$cont1 \leftarrow 0$
repeat
  $p1 \leftarrow points[currPoint]$
  $cont \leftarrow 0$
  for $k = 1$ **to** $NumPoints$ **do**
    if $points[k].used =$ FALSE **then**
      if $cont = 0$ **then**
        $p2 \leftarrow points[k]$
        $minAngle \leftarrow k$
        $cont++$
      else if $findAngle(p1, points[k]) \leq findAngle(p1, p2)$ **then** ▷ find minimum angle
        if $findAngle(p1, points[k]) = findAngle(p1, p2)$ **then** ▷ check for collinearity
          if $distance(p1, points[k]) \leq distance(p1, p2)$ **then**
            $points[k].used \leftarrow$ TRUE
            step
          else
            $points[minAngle].used \leftarrow$ TRUE
          end if
        end if
        $p2 \leftarrow points[k]$
        $minAngle \leftarrow k$
      end if
    end if
  end for
  if $cont1 = 0$ **then**
    $cont1++$
    $points[minPoint].used \leftarrow$ FALSE
  end if
  $points[minAngle].used \leftarrow$ TRUE
  $currPoint \leftarrow minAngle$
  Call addConvexHullPoint ($currPoint$)
until $currPoint = minPoint$
End

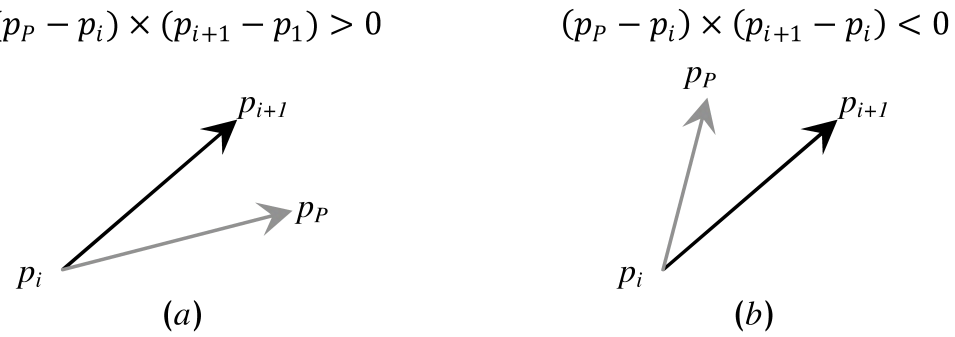


Figure 5.8: Point-in-polygon determination

Algorithm 3 *Point-in-Polygon* (Point P, Convex Hull S)

Input: Let P, a point with coordinates x_p, y_p, be the application point of force F_R on box b. Let S be a vector of n points with coordinates x_i, $y_i (i = 1, ..., n)$, representing the support polygon of box b.

Output: Let *stable* be a boolean value where FALSE represents a point outside the polygon S and TRUE the opposite case.

Begin
stable ← TRUE
for $i = 1$ **to** $n - 1$ **do**
  vectorproduct ← (x_p - x_i) * (y_{i+1} - y_i) − (x_{i+1} - x_i) * (y_p - y_i)
  if *vectorproduct* > 0 **then**
    stable=FALSE
    return stable
  end if
end for
return stable
End

Determine reaction forces

The determination of the reaction forces of all the support boxes of box b starts by writing equations (5.1) and (5.2) as three scalar equations (5.3). Only three scalar equations are used since there are no forces with a component in both the x and y-axis and as consequence no moments about the z-axis are generated.

$$\begin{aligned} \sum F_z &= 0 \\ \sum M_x &= 0 \\ \sum M_y &= 0 \end{aligned} \tag{5.3}$$

When the box b equation system is statically indeterminate, i.e., if the number of unknowns in (5.3) is greater than three, additional equations obtained from geometry deformation conditions are added. For this purpose box b is modelled as a rigid body while the support boxes are modelled as deformable bodies with the same linear elasticity. By considering the box b a rigid body, the base of the box forms a plane in space with equation (5.4) and is used as a compatibility condition between reaction forces.

$$z = \alpha x + \beta y + z_0 \tag{5.4}$$

The reaction forces of the support box are then treated as springs with equal stiffness. By choosing one of the support boxes as the origin (z_0), a displacement relation can be established with each of the remaining support boxes, thereby obtaining the compatibility conditions required to solve the problem. This method is used in the field of mechanics of materials and is referred to as the "Force Method of Analysis" (Hibbeler, 1994).

5.4 Loading/unloading operations efficiency

When defining a cargo arrangement, a high utilization of the container space is important for achieving high efficiency in the transportation system. However, the expected increase in cargo transportation in coming years will also require that other elements of the supply chain, such as warehouses and distribution centres, improve their performance in order to avoid bottlenecks in the supply chain, since the cargo throughput of those points of the supply chain is also expected to increase.

Reducing container cargo loading and unloading time will contribute to reduced bottlenecks in warehouse inbound and outbound operations. (Un)loading operations are usually measured by the number of (un)loaded boxes per man-hour. With effective (un)loading operations, the consequent increase in the number of boxes per man-hour will impact on the performance of the supply chain. It is then desirable that the cargo arrangement in the container is influenced by the effectiveness of the (un)loading operations.

There are several techniques for (un)loading cargo from containers or trailers:

- Manual handling - manual (un)loading of cargo done by workers by lifting and carrying boxes to/from pallets or a flexible conveyor system.
- Using handling equipment - (un)loading by means of forklift trucks, pallet trucks or other mechanical handling equipment.
- Automated systems - handling systems which (un)load cargo from/to containers, trucks or trailers automatically. This can be achieved by using conveyors, rollers, skates or robot arms.

From an ergonomic perspective, manual handling activities can be physically very challenging and have higher probability of injury for the workers. This is mainly due to muscular fatigue, repetitious movements like bending, lifting or twisting, carrying loads for a distance and overhead reaching and lifting (Bhattacharya and McGlothlin, 2012). There are several types of equipment designed to help in lifting boxes which can mitigate the stress that operators are subjected to during manual (un)loading operations.

As with manual loading, these systems require that there is no obstruction when loading, either vertical, i.e., along the z-axis, or horizontal, parallel to the x-axis. When a new box b is placed inside container C, a check is made for any obstruction by detecting whether the empty space required to load the box has not been partially filled by any of the boxes already placed. A box b located at (x_b, y_b, z_b) and with dimensions (d_b, w_b, h_b) can be inserted in container C, loaded with boxes b_j, if condition (5.5) is verified. It considers that if all placed boxes, b_j, are located behind, to the right, to the left or underneath the final position for box b then definitively there are no obstructions to placing box b.

$$(x_j + d_j \leq x_b) \vee (y_b + w_b \leq y_j) \vee (y_j + w_j \leq y_b) \vee (z_j + h_j \leq z_b) \tag{5.5}$$

Condition (5.6) must also be satisfied for those boxes not yet loaded. This condition guarantees that a box b can be loaded if all boxes b_j that are located under box b are already in place, i.e., all the remaining boxes b_i to be loaded are to be located behind, in front, to the right, to the left or above box b.

$$(x_i + d_i \leq x_b) \vee (x_b + d_b \leq x_i) \vee (y_b + w_b \leq y_i) \vee (y_i + w_i \leq y_b) \vee (z_i \geq z_b + h_b) \tag{5.6}$$

Another relevant aspect to the (un)loading operations is the method by which the boxes are stacked in the container. The two most common loading methods are the layer-by-layer and the pyramiding methods. The pyramiding method consists of (un)loading boxes in a diagonal pattern from the back-bottom to front-top. In the layer-by-layer method boxes are (un)loaded one layer at a time within reachable arm's-length (see Figure 5.9).

Based on the existing different (un)loading methods it was decided that the loading sequence would follow a wall building approach, where each box is placed inside the container, starting in the back-bottom-left corner, with boxes placed sequentially along the y-axis and then upward along the z-axis, as illustrated in Figure 5.10. This method keeps the box closer to the worker's body prior to lifting or bending, and helps reduce the stress on the shoulders, back, and possibly knees during the reaching and pulling motions, which also contributes to the efficiency of the operation.

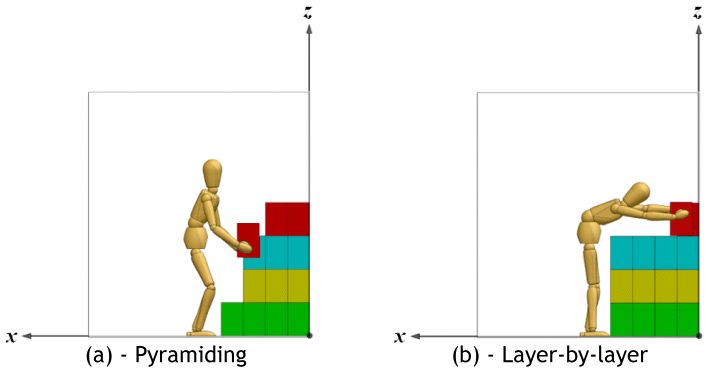


Figure 5.9: Stacking methods of container loading

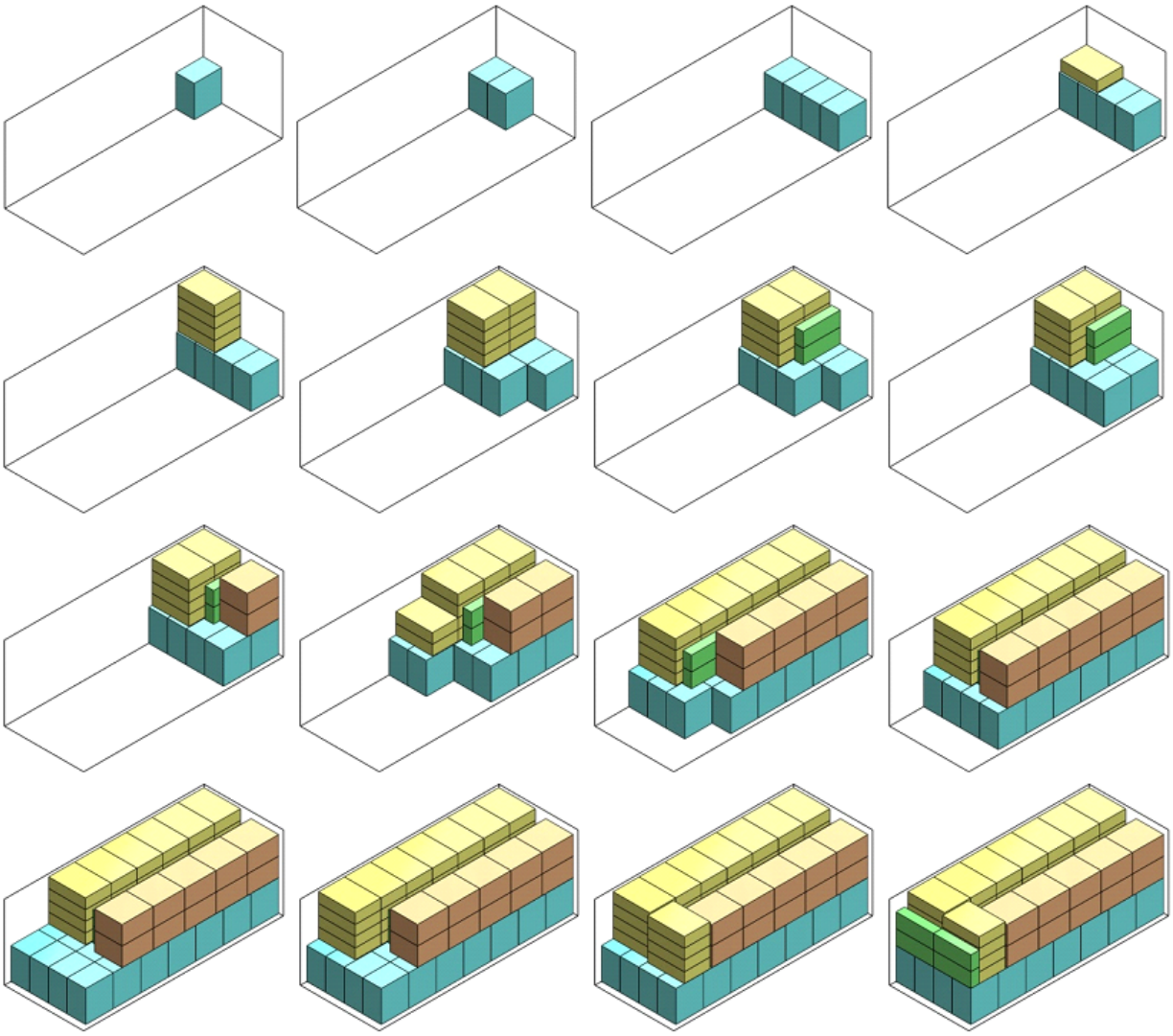


Figure 5.10: Example of a cargo loading sequence

5.4.1 The loading efficiency metrics

To evaluate the efficiency of the loading operation, the average arm's-length of the loaded boxes is used as a metric. The *arm's-length* of box b can be determined by the difference between x_1, defined as the maximum x-axis coordinate of a set of boxes N located beneath b, and x_2, the maximum x-axis coordinate of box b, such that $arm's\text{-}length = max(x_1 - x_2, 0)$. Let N be determined by all boxes b_j placed beneath b such that:

$$(x_b < x_j + d_j) \wedge (x_j < x_b + d_b) \wedge (y_b < y_j + w_j) \wedge (y_j < y_b + w_b) \wedge (z_b \geq z_j + h_j)$$

This reflects the effort made by the operator to load a complete cargo arrangement. Figure 5.11 presents an illustration of the *arm's-length* metric. The *arm's-length* value is 0 if there is an overhang in the x-axis direction. There are several other issues related to loading efficiency, as for example the loading weight of the boxes to be lifted, overhead reach, etc. However, since other constraints, such as load bearing, are not considered in the problem, it was decided that the *arm's-length* metric would be used, since it is a geometric efficiency metric.

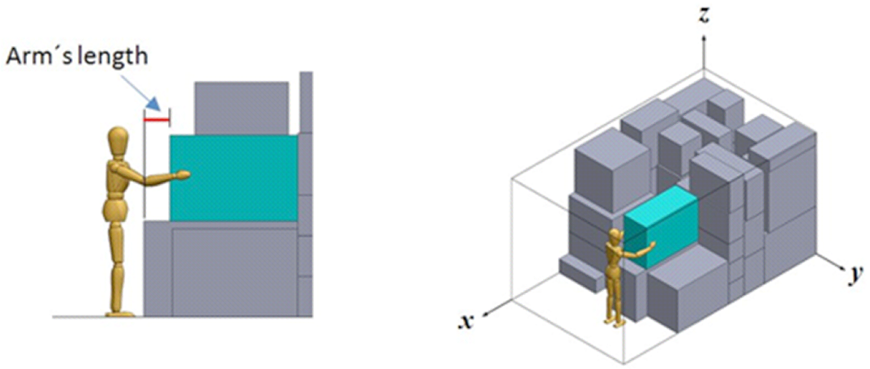


Figure 5.11: *arm's-length* metrics

5.5 The Physical Packing Sequence Problem

The Physical Packing Sequence Problem (PPSP) uses a loading arrangement M, generated by a container loading algorithm, to determine a sequence by which each box $b_i (i = 1, ..., n)$ with depth (d_i), width (w_i) and height (h_i) is placed in a predetermined position (x_i, y_i, z_i) inside a container C with depth (D), width (W) and height (H), while at the same time the following additional loading constraints are considered:

- Static stability (Hibbeler, 2010b) - Defined as a state where a box is not accelerating in any direction. A box which is currently in equilibrium will remain in equilibrium if two conditions are met:
  - the vector sum of forces applied on the box is zero and;

– the vector sum of torques that act to rotate the box is zero.

- Loading operation efficiency - The boxes are loaded following a wall building approach, starting in the back-bottom-left corner of the container while guaranteeing that every box is loaded without vertical and horizontal obstructions, all of its support boxes have been loaded and the arm's-length of the box does not surpass a predefined value.

5.5.1 The Physical Packing Sequence Algorithm

The Physical Packing Sequence Problem discussed in this paper is solved considering the static stability approach and the loading operation efficiency criteria previously described. The proposed algorithm is presented in Algorithm 4.

The algorithm starts by ordering the set of all the boxes to load M, by non-decreasing order of their x, z and y placement coordinates, respectively. This ordering corresponds to a *wall building* strategy. The relative position of all boxes can subsequently be found. This allows the support box to be found for each box, and the set of boxes H_i located beneath and behind can be determined for each box i such that

$$\begin{cases} x_i < x_j + d_j \\ x_j < x_i + d_i \\ y_i < y_j + w_j \\ y_j < y_i + w_i \\ z_i \geq z_j + h_j \end{cases} \quad \vee \quad \begin{cases} y_i < y_j + w_j \\ y_j < y_i + w_i \\ z_i < z_j + h_j \\ z_j < z_i + h_i \\ x_i \geq x_j + d_j \end{cases}$$

Starting with the first box on list M, a box i is selected and it is checked if any of the boxes located beneath and behind are in M, i.e., if $H_i \cap M = \emptyset$. If that is not the case, the next box is selected. Then, for the selected box, the *arm's-length* is calculated. If the *arm's-length* is within the *arm's-length* limits, the static stability of the box is checked using the static stability algorithm, otherwise the box is removed from the list and a new box is selected. If the box is stable, it is loaded in the container. This procedure is iteratively repeated until there are no boxes in the list. A loading arrangement is considered feasible if all the boxes are loaded in the container.

5.6 Numerical experiments

In this section the results of a set of experiments run to evaluate the impact of the Physical Packing Sequence Algorithm are reported. The algorithm was coded in C and was run on an Intel Core2 Duo at 2.20 GHz computer with 4 Gbytes of RAM.

5.6.1 Test problem instances

To the best of our knowledge there are no test instances published for the PPSP. For this reason we propose 2 sets of 15 classes of test instances, with a total of 100 cargo arrangement

Algorithm 4 Physical Packing Sequence (Loading arrangement M)

Input: Let M be a set of n boxes b_i with dimensions d_i, w_i, h_i, positioned at coordinates x_i, y_i, z_i, $i = 1, ..., n$

Output: let k_i be the order position of box b_i, $i = 1, ..., n$

Begin
Sort M in non decreasing order of coordinates x, z, y;
$order\_position \leftarrow 1$
Let H_i be the set of boxes beneath and behind box b_i
while $(M \neq \emptyset)$ **do**
  for $i = 1$ **to** n **do**
    if $b_i \in M$ **then** ▷ Is the box waiting to be loaded?
      if $H_i \cap M = \emptyset$ **then** ▷ All boxes beneath and behind have been processed?
        remove b_i from M
        $arms_i \leftarrow$ **Call** arms_lenght(b_i) ▷ Determine the *arm's-length* of box b_i
        if $arms_i \leq arms\_limit$ **then** ▷ Is the *arm's-length* acceptable?
          if (**Call** Static_Stability(b_i)) **then** ▷ Is the cargo stable?
            $k_i \leftarrow order\_position$
            $order\_position++$
            break
          end if
        end if
      end if
    end if
  end for
end while
End

solutions per class. These are solutions of the well-known instances of Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999) provided by the algorithm of Parreño et al. (2008). One of the sets was obtained from Parreño et al. (2008) by enforcing full base support for all boxes, while for the other set the full base support constraint was not considered.

5.6.2 Computational results

Table 5.1 shows, for each class of instances, the average results for the 100 arrangements, when the PPSA is applied to container loading solutions. The table presents the average percentage of boxes loaded by the algorithm and the average *arm's-length* to which the worker was effectively exposed, in cm, for the 2 sets of solutions previously mentioned. For each set, results were obtained with or without the maximum *arm's-length* constraint. A value of 80 cm was considered for this maximum. For the set of instances where full base support was not enforced, the average percentage of boxes was calculated that would be

loaded when a minimum of 85% support for the base of the box was set as a requirement, instead of the stability criterion proposed in this paper.

It was found that if there is no *arm's-length* constraint and boxes are fully supported, it is always possible to find a physical packing sequence for the cargo arrangements. It can also be observed that when the *arm's-length* constraint is enforced, the impact is similar across instances, with a decrease of around 3% in the feasibility of the loading arrangements.

The results for the test instances where full base support was not imposed, and without the *arm's-length* constraint, show that an average of around 82% of the boxes were loaded. A decrease in the average percentage of loaded boxes can be observed with the increase in box heterogeneity (which increases from BR1 to BR15). This decrease can also be observed when the *arm's-length* constraint is enforced: in this case the average percentage of loaded boxes decreases from 95% to 62%.

If the stability criterion proposed in this paper was replaced by the partial base support constraint, where at least 85% of the base of the box must be supported, on average only 68% of the boxes would be loaded. This provides evidence that the proposed stability algorithm is less restrictive than usual static stability measures, while fully guaranteeing cargo static stability.

5.7 Concluding remarks

In this paper we have proposed two algorithms, a Static Stability Algorithm based on the static mechanical equilibrium conditions for rigid bodies, and a Physical Packing Sequence Algorithm to evaluate the feasibility of cargo arrangements generated by existing Container Loading Problem algorithms. The new proposed Static Stability Algorithm was used within the Physical Packing Sequence Algorithm, demonstrating its ability to be combined with other algorithms as an evaluation function. Some of the major benefits of the algorithm include its short processing time, the ability to evaluate static stability when the supported area of a box is below 50%, and its proximity to real-world static stability. This capability can lead to the development of new strategies for the generation of cargo arrangements in the Container Loading Problem.

The computational experiments demonstrate that it is a less restrictive approach to stability than the thumb rules and measures usually found in the literature. For the test instances where full base support is not enforced on the Container Loading Problem solutions, this algorithm considers that an average of only 82% of the boxes can be loaded safely and stably; when *arm's-length* is limited to a maximum of 80 cm only 76% of the boxes can actually be loaded without stability issues.

Table 5.1: Computational experiments

		no arm's-length constraint			arm's-length $\leqslant$ 80 cm		
	Parreño et al. Instances	% Loaded Boxes	Arm's Length (cm)	% of boxes with at least 85% of base support	% Loaded Boxes	Arm's Length (cm)	% of boxes with at least 85% of base support
Full Base Support	**BR1**	100%	8.5	-	99%	8.0	-
	BR2	100%	13.5	-	98%	12.2	-
	BR3	100%	11.5	-	98%	10.0	-
	BR4	100%	13.3	-	98%	11.8	-
	BR5	100%	12.9	-	98%	11.3	-
	BR6	100%	13.7	-	97%	11.4	-
	BR7	100%	14.4	-	96%	11.8	-
	BR8	100%	16.2	-	97%	13.7	-
	BR9	100%	15.5	-	96%	13.1	-
	BR10	100%	15.7	-	95%	12.5	-
	BR11	100%	15.1	-	95%	11.9	-
	BR12	100%	14.6	-	95%	11.4	-
	BR13	100%	13.0	-	97%	10.7	-
	BR14	100%	12.3	-	97%	10.3	-
	BR15	100%	12.0	-	97%	10.1	-
	Mean	**100%**	**13.5**		**97%**	**11.3**	
Without Full Base Support	**BR1**	97%	16.1	96%	95%	14.5	91%
	BR2	95%	16.2	90%	92%	13.9	87%
	BR3	92%	19.0	84%	88%	15.9	81%
	BR4	92%	21.1	83%	88%	17.8	79%
	BR5	89%	21.2	79%	83%	16.7	75%
	BR6	88%	21.8	76%	82%	17.4	72%
	BR7	85%	23.3	73%	79%	18.3	68%
	BR8	80%	26.5	65%	72%	20.2	60%
	BR9	75%	26.0	60%	68%	19.9	55%
	BR10	74%	27.4	57%	66%	20.4	52%
	BR11	71%	27.1	53%	63%	20.1	49%
	BR12	70%	27.2	52%	62%	20.2	48%
	BR13	70%	27.0	50%	62%	20.3	46%
	BR14	68%	25.3	48%	62%	19.3	45%
	BR15	69%	24.1	48%	62%	18.0	45%
	Mean	**82%**	**23.3**	**68%**	**76%**	**18.2**	**64%**

Chapter 6

A Container Loading Algorithm with Static Mechanical Equilibrium Stability Constraints

A. Galrão Ramos[†] [‡], José F. Oliveira[†], José F. Gonçalves[§] and Manuel P. Lopes[‡]

[†] INESC-TEC and Faculty of Engineering, University of Porto
[‡] CIDEM, School of Engineering, Polytechnic of Porto
[§] LIAAD, INESC-TEC and Faculty of Economics, University of Porto

Abstract

The Container Loading Problem (CLP) literature has traditionally guaranteed cargo static stability by imposing the full support constraint for the base of the box. Used as a proxy for real-world static stability, this constraint excessively restricts the container space utilization and has conditioned the algorithms developed for this problem. In this paper we propose a container loading algorithm with static stability constraints based on the static mechanical equilibrium conditions applied to rigid bodies, which derive from Newton's laws of motion. The algorithm is a multi-population biased random-key genetic algorithm, with a new placement procedure that uses the *maximal-spaces* representation to manage empty spaces, and a layer building strategy to fill the *maximal-spaces*. The new static stability criterion is embedded in the placement procedure and in the evaluation function of the algorithm. The new algorithm is extensively tested on well-known literature benchmark instances using three variants: no stability constraint, the classical full base support constraint and with the new static stability constraint—a comparison is then made with the state-of-the-art algorithms for the CLP. The computational experiments show that by using the new stability criterion it is always possible to achieve a higher percentage of space utilization

than with the classical full base support constraint, for all classes of problems, while still guaranteeing static stability. Moreover, for highly heterogeneous cargo the new algorithm with full base support constraint outperforms the other literature approaches, improving the best solutions known for these classes of problems.

6.1 Introduction

The Container Loading Problem (CLP) is a real-world driven, combinatorial optimization problem that addresses the optimization of the spatial arrangement of cargo inside containers or transportation vehicles, maximizing the usage of space.

As an assignment problem, it can have two basic objectives: the maximization of the value of the cargo loaded, when the number of containers is not sufficient to accommodate all the cargo, or the minimization of the value of containers, when there are sufficient containers to accommodate all the cargo.

The problem belongs to the wider combinatorial optimization class of Cutting and Packing problems. According to the typology defined by Wäscher et al. (2007) for Cutting and Packing problems, these can be classified according to dimensionality, assortment of large items, assortment of small items, kind of assignment and shape of small items. In this paper we will consider two types of problem with an output maximization objective. These problems can be classified either as three-dimensional, rectangular single large object placement problems (3D-SLOPP) or as three-dimensional, rectangular single knapsack problems (3D-SKP), depending on the cargo heterogeneity.

The problem is highly relevant to the field of transport management since it impacts customer satisfaction, operational efficiency and transport safety. The arrangements for loading cargo into containers should comply with various requirements: cargo should not become damaged during transportation, transportation space should be used efficiently and workers' safety should not be breached during loading and unloading of cargo.

However, if the approach to the problem does not consider real-world constraints, such as cargo stability, container weight-limit or cargo orientation constraints, the solution will be of limited applicability to real-world scenarios. Cargo stability is considered in the literature as one of the most important CLP constraints. Its impact is not confined to the cargo as it can also influence the safety of both workers involved in loading operations and other persons or vehicles during transportation.

In the CLP literature, cargo stability has been guaranteed by imposing the full support constraint on the base of the boxes. Although guaranteeing static stability, it excessively restricts the container space usage and does not necessarily meet real-world needs when overhanging cargo is allowed.

The CLP addressed in this work can be stated as follows: A given set of small items of parallelepiped shape of type $k(k = 1, ..., K)$ (known as boxes), $B = b_1, b_2, ..., b_K$, where each box type, in quantity n_k, is characterized by its depth, width and height (d_k, w_k, h_k) are to be loaded into a large object of parallelepiped shape (known as a container), C, characterized by its depth, width and height, (D, W, H), with the objective of achieving a

maximum utilization of the volume of the container, while meeting the following geometric loading constraints:

- each face of a box must be parallel to one of the faces of the container;
- there must be no overlap between the boxes;
- all boxes must lie entirely within the container;
- each box must be placed according to one of its possible orientations—each box type can have up to six possible orientations.

The mechanical properties of the container and the boxes also necessitate the following additional practical constraints:

- boxes can only be loaded through the container entrance;
- static stability—each box must be able to maintain its loading position undisturbed during cargo loading;
- all boxes are rigid;
- the centre of gravity of each box is assumed to be its geometric centre.

The dimensions (D, W, H) of container C lie parallel to the x, y and z axes, respectively, of the first octant of a Cartesian coordinates system, with the back-bottom-left corner lying at the origin of the coordinates system. The placement of a box b_i in the container is given by its minimum and maximum coordinates, (x_{1i}, y_{1i}, z_{1i}) and (x_{2i}, y_{2i}, z_{2i}), respectively.

The aim of this work is to present an algorithm for the CLP that addresses cargo stability under a realistic framework. The proposed algorithm combines a multi-population biased random-key genetic algorithm with a constructive heuristic that enforces a static stability constraint based on the static mechanical equilibrium conditions applied to rigid bodies, which derive from Newton's laws of motion. The constructive heuristic uses a *maximal-spaces* representation to manage empty spaces, and a layer approach for filling the *maximal-spaces*.

The remainder of the paper is organized as follows. Section 6.2 presents an overview of the literature covering the CLP and static stability within the CLP. In Section 6.3 the Container Loading Algorithm with Static Stability is presented. Section 6.4 reports the results from the computational experiments. Finally, Section 6.5 draws some conclusions from the findings.

6.2 Literature review

Many approaches have been proposed for solving the 3D-SLOPP and the 3D-SKP. The number of exact methods proposed is very limited and these can only solve problems of

limited size. Exact methods were developed by Padberg (2000), Fekete et al. (2007), Junqueira et al. (2012b) and Junqueira et al. (2012a). Alternatively, other methods have been proposed to find near-optimal packing solutions. Fanslau and Bortfeldt (2010) classified these methods as conventional heuristics, metaheuristics and tree-search methods. Conventional heuristics include construction heuristics and improvement heuristics. Construction heuristics derive a single feasible solution that is directly employed or used as a starting point for local search heuristics. They do not attempt to improve the obtained solution. Improvement heuristics try to iteratively improve the solution of already known solutions. Examples include methods developed by George and Robinson (1980), Bischoff et al. (1995), Lim et al. (2003) and Liu et al. (2011a). Metaheuristics can be seen as general-purpose methods that aim to effectively and efficiently explore a search space using intensification (exploitation) and diversification (exploration) strategies. According to the philosophy followed, metaheuristics can be seen as extended variants of improvement heuristics that aim to escape from a local optimal solution and continue with the exploration of the search space with the expectation of finding a better solution, or as a population-based approach where the search space is explored in each iteration by a population. Examples of such approaches can be found in the Genetic Algorithms of Bortfeldt and Gehring (2001) and Gonçalves and Resende (2012), in the use of Tabu Search by Bortfeldt et al. (2003) and Liu et al. (2011a) and the greedy randomized adaptative search procedures (GRASP) of Moura and Oliveira (2005) and Parreño et al. (2008). Tree-search methods include tree-search and graph-search methods. These are methods that can be used when the set of all feasible solutions of the optimization problem can be represented by a tree or a graph. Examples include the works of Fanslau and Bortfeldt (2010), Zhu and Lim (2012) and Araya and Riff (2014).

A common feature of these methods is that they search over a representation or codification of the solution (usually box sequences) and therefore require a constructive box loading heuristic to generate a feasible solution. Usually, the heuristic iteratively selects a location inside the container and a box (or set of boxes) to place at that location, until no more locations or boxes are available. Both of these decisions are related to the way the empty space of the container is managed (and therefore the way in which all potential placement locations are evaluated) and the way box arrangements are generated.

Spatial representation Different approaches can be found in the literature to managing empty spaces. Ngoi et al. (1994) use a single three-dimensional matrix representation of objects and empty spaces as a combination of variable orthorhombic cells. Each three-dimensional matrix is composed of a chain of two-dimensional matrices that represent the details of horizontal layers of constant thickness. Bischoff (2006) proposed an adaptation of the Ngoi et al. (1994) representation that does not involve the creation of the horizontal layer matrices. Instead, a single two dimensional matrix that represents a view from the top of the container is required. Another approach was proposed by George and Robinson (1980). Their methodology is based on the assumption that after the placement of a box in a packing space, the remaining unused space opens up three new spaces. The three

spaces, illustrated in Figure 6.1, are created in the following order: spare depthwise space, spare widthwise space and spare heightwise space. Each space is therefore represented by its depth, height and width and the coordinates of its rear-left-bottom vertex.

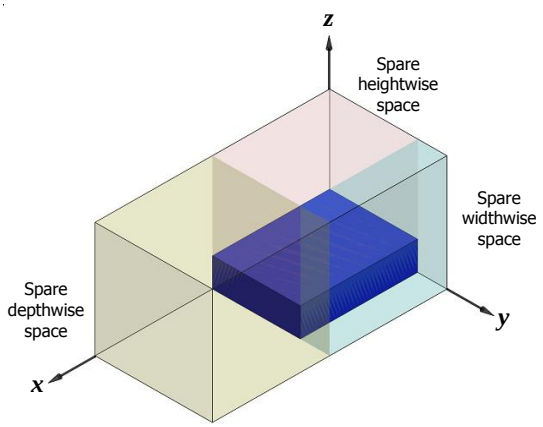


Figure 6.1: George and Robinson (1980) empty space subdivision

The approach proposed by George and Robinson (1980) only considered one variant to partitioning the empty space. Other authors, such as Bortfeldt et al. (2003) and Fanslau and Bortfeldt (2010), later extended the George and Robinson (1980) approach and considered other variants of the empty space subdivision, as illustrated in Figure 6.2. In these approaches, empty spaces are represented as a set of disjoint spaces.

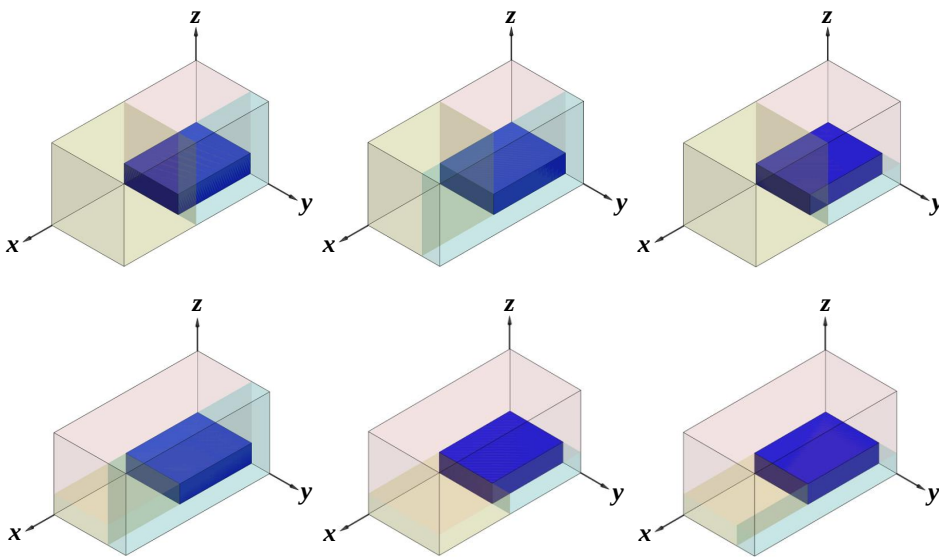


Figure 6.2: Variants of the empty space subdivision

For the two and three-dimensional cutting stock problem, Lai and Chan (1997) proposed a representation of empty spaces as a set of non-disjoint empty spaces. These empty spaces

have the largest parallelepiped shape that can be considered and are managed using the "Interval Generation" procedure. However, this procedure was not applied, by the authors, to the three-dimensional CLP. Later Parreño et al. (2008) used this representation in the CLP and designated the non-disjoint empty spaces as *maximal-spaces*. The *maximal-spaces* representation is illustrated in Figure 6.3. A *maximal-space* s representation is given by its minimum and maximum coordinates, (x_{1s}, y_{1s}, z_{1s}) and (x_{2s}, y_{2s}, z_{2s}), respectively, and an Insertion Vertex V, (that is, the vertex of the *maximal-space* where the boxes will be packed).

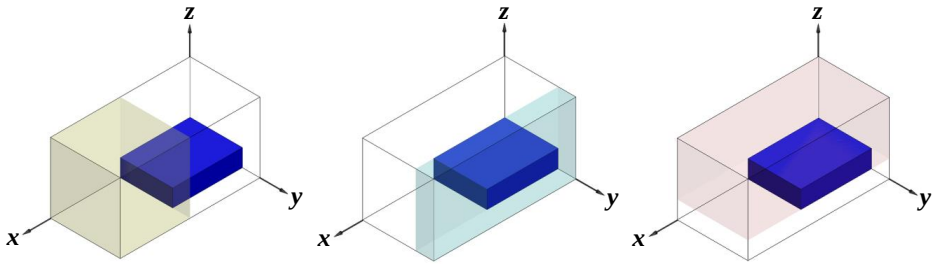


Figure 6.3: Maximal-space representation

Box arrangement strategy Either single box or multiple box strategies can be followed at each placement; each iteration of the former places only one box inside the container, each iteration of the latter places a set of boxes together. In a multiple box strategy the arrangements can be formed from sets of identical or non-identical boxes. Multiple boxes can also be arranged by dimensionality, (that is, one, two and three dimensions). In the one dimensional arrangement, boxes are grouped along a single axis, and therefore generate a column. In a two dimensional arrangement, boxes are grouped in a plane, (that is, along two axes, generating layers). In the three dimensional arrangement, boxes are grouped along the three axes, generating blocks.

The constructive heuristic results from the combination of the selected spatial representation and box arrangement strategy and a set of rules that select location and box arrangement at each iteration.

The best existing CLP algorithm that uses a two dimensional arrangement strategy is the multi-population biased random-key genetic algorithm developed by Gonçalves and Resende (2012), while the best three dimensional arrangement strategy is the Beam Search Approach developed by Araya and Riff (2014).

6.2.1 Static stability

A recent literature review of container loading constraints by Bortfeldt and Wäscher (2013) highlights the particular relevance placed on stability constraints. Considered to be one of the most important CLP constraints, stability has been addressed by a large number of authors usually following a rather simplified approach that often considers the term "stability" as if it was self-explanatory. Concepts, such as loading stability and transportation

stability, are sometimes not addressed separately, but there are some approaches to stability found in the literature that make a clear distinction between static (vertical) and dynamic (horizontal) stability, (that is, the stability of cargo during loading operations and the stability of cargo during transportation). However, the majority of approaches only focus on static stability (Bortfeldt and Wäscher, 2013).

The approaches to static stability found in the CLP literature can be classified according to the type of stability constraint to be enforced: full base support, partial base support or static mechanical equilibrium. Of these, full base support and static mechanical equilibrium both guarantee static stability, while partial support does not.

- **Full base support** requires the entire base of a box be in contact with the base of the container or with the top surface of other boxes. As a result, no overhanging boxes are allowed. Examples can be found in Bischoff and Ratcliff (1995a), Bortfeldt and Gehring (2001), Gonçalves and Resende (2012) and Zhu and Lim (2012).

- **Partial base support** requires that either the entire base of a box be in contact with the base of the container, or a pre-specified percentage of the area of a box base be in contact with the top surface of other boxes, thereby allowing overhanging. As an example, Carpenter and Dowsland (1985) require the contact area to fall in the range of 95% to 75%, while Christensen and Rousøe (2009) require a minimum of 80%, Gendreau et al. (2006), Fuellerer et al. (2010) and Tarantilis et al. (2009), 75%, Gehring and Bortfeldt (1997), 70% and Mack et al. (2004), 55%.

- **Static mechanical equilibrium** requires that the entire base of a box be in contact with the base of the container or,

  - the sum of external forces acting on the box is zero and;
  - the sum of torque exerted by the external forces is zero.

  An example, applied to the three-dimensional bin packing problem, can be found in de Castro Silva et al. (2003). The center of gravity condition is a condition found in the literature which is derived from the static mechanical equilibrium conditions applied to rigid bodies. This condition requires the center of gravity of a box be located above the contact surface of the supporting boxes (Mack et al., 2004; Lin et al., 2006). However, by itself, enforcing this condition does not guarantee static stability (Ramos et al., 2014b).

Recent approaches in the literature consider the concept of static stability as equivalent to enforcing full base support (Gonçalves and Resende, 2012; Zhu and Lim, 2012; Araya and Riff, 2014). Consequently these approaches developed both algorithms that enforce full base support and algorithms that have no such requirement (Unsupported). The performance benchmarking used for these algorithms is the percentage of volume loaded with or without full base support. As a result, the goal is not to obtain a loading arrangement that is statically stable but a loading arrangement where all boxes have full base support.

Table 6.1 summarizes the results of the previously identified best existing CLP approaches without the static stability constraint (Unsupported) and with enforcement of the full base support constraint (Full Support). In the table BSG-CLP refers to the Beam Search Approach of Araya and Riff (2014) and BRKGA refers to the multi-population biased random-key genetic algorithm of Gonçalves and Resende (2012). The values in columns 2 to 5 of the table correspond to the average percentage of volume utilization for the test instances of Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999) organized in 15 classes, with a total of 100 instances per class. Classes BR1 to BR7 are weakly heterogeneous classes while BR8 to BR15 are strongly heterogeneous classes. The values in the columns below the label "Difference", represent the difference between the results of the Unsupported and Full Support variants. Looking at the results leads to the conclusion that the full base support constraint is very costly for algorithm efficiency, particularly in the strongly heterogeneous instances. To give an idea of the impact this represents for transportation cost, 3% of the volume of a 40-foot container is around 1.5m^3 of space.

Table 6.1: Comparison of the best existing algorithms using Full Support and Unsupported variants

	Full Support (FS)		**Unsupported (U)**		**Diference (U-FS)**	
Problem	**BSG**	**BRKGA**	**BSG**	**BRKGA**	**BSG**	**BRKGA**
BR 1-7	94,74	94,53	96,11	95,74	1,36	1,20
BR 8-15	91,22	90,23	94,78	93,49	3,56	3,26
BR 1-15	92,87	92,24	95,40	94,54	2,53	2,30

The best performing algorithms use two different heuristic approaches: BSG-CLP uses a beam search heuristic approach while BRKGA uses the genetic algorithm. The BSG-CLP provides the best results for the Unsupported and Full Support CLP variants. However, these algorithms are not so flexible when faced with additional constraints, such as load bearing, weight limit or weight distribution, that can only be evaluated after the loading arrangement has been completed.

6.3 The container loading algorithm with static stability

The proposed container loading algorithm hybridizes a multi-population biased random-key genetic algorithm (BRKGA) with a constructive heuristic embedded with a static stability constraint, based on the static mechanical equilibrium conditions applied to rigid bodies, which derive from Newton's laws of motion. In a BRKGA, as in all metaheuristics, there is a problem-independent part, responsible for the evolution of coded solutions (chromosomes), and a problem-specific part, responsible for decoding the chromosome, generating a solution and evaluating its fitness (Gonçalves and Resende, 2011).

The use of a BRKGA for the CLP was first proposed by Gonçalves and Resende (2012)

to solve the 3D-SLOPP and 3D-SKP and by (Gonçalves and Resende, 2013) for the three-dimensional, Single Bin-Size Bin Packing Problem (3D-SBSBPP).

Gonçalves and Resende (2011) stated that it is the chromosome representation and the problem-specific part of the algorithm that requires research effort, and distinguishes the different BRKGA's approaches to the same or different problems. For the here proposed algorithm, a new chromosome representation and a new problem-specific part (that is, a constructive heuristic) are specified. Figure 6.4 illustrates the architecture of the algorithm.

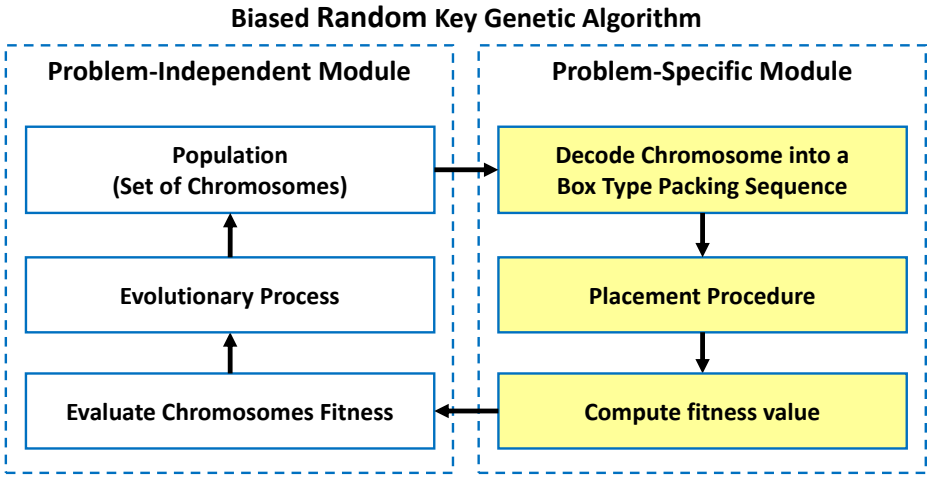


Figure 6.4: Architecture of the base algorithm

The constructive heuristic starts with the decoding procedure that generates the sequence by which the boxes are loaded into the container. The generation of an actual solution is obtained by a box placement procedure that uses a *maximal-space* representation of the empty spaces and a layer building strategy (2D box arrangement strategy). The placement procedure follows four main steps:

- Step 1: Selecting the box type
- Step 2: Selecting the *maximal-space*
- Step 3: Filling the *maximal-space*
- Step 4: Updating the system information

Finally, the value of the solution is determined by the percentage of total packed volume.

The following subsection firstly provides a description of two indirect solution representations and decoding procedures, and then presents a placement heuristic and the fitness function for the Unsupported and Full Support CLP variants. Finally the description of the placement heuristic and fitness function for the static mechanical equilibrium variant, used to guarantee cargo static stability, is presented.

6.3.1 Chromosome encoding and decoding

A chromosome in a genetic algorithm represents a solution to the problem. It can be a direct or indirect representation, depending on whether the chromosome is a representation of the solution to the original problem or whether additional procedures are needed to obtain a solution. The use of an indirect representation for the CLP, within a genetic algorithm framework, is preferable to the direct use of chromosomes as packing sequences for the CLP, since in a direct representation, the genes must represent the box placement coordinates, which would make the overlay constraints much harder to enforce. By using a biased random-key genetic algorithm, where the chromosome is encoded as a vector of random keys (real numbers between 0 and 1), an indirect representation of the solution is used which guarantees that the offspring formed by crossover of the chromosomes are feasible solutions (Gonçalves and Resende, 2011). The use of an indirect representation of the solution requires the chromosomes to be decoded for the CLP.

Two indirect representations of the solution are adopted: one strongly heterogeneous and one weakly heterogeneous. In the strongly heterogeneous case, each gene is used to represent the box type for each of the boxes to be loaded. In the weakly heterogeneous case, a first set of genes represent the box type of each of the boxes to be loaded and a second set of genes represents one of the three planes along which layers of boxes can be built up ($x-y$, $x-z$ and $y-z$) to fill the *maximal-spaces*. The second set of genes aims increasing the diversity of generated solutions by the constructive heuristic.

Let chromosome $G = \{gene_1, gene_2, ..., gene_g\}$ be an indirect representation of a CLP solution and let M represent the number of boxes to be loaded ($M = \sum_{k=1}^{K} n_k$). The number of genes, g, of the chromosome depends on the indirect solution representation and the total number of boxes to be loaded M.

- In the **strongly heterogeneous** representation, g is equal to M. Each $gene_j$ represents the type of box of the j^{th} box to be loaded. By sorting the genes in ascending order a new box type sequence is generated. The new vector sequence is designated the Box Type Packing Sequence (BTPS).

- In the **weakly heterogeneous** representation, g is equal to $2M$. For the first M genes, each $gene_j$ represents the type of box of the jth box to be loaded while the last M genes ($gene_{M+j}$) represent the layer filling plane of the j^{th} box to load. The layer filling plane of box j is determined using (6.1).

$$\begin{cases} 0 \leq gene_{M+j} < 1/3 & x-y \text{ plane} \\ 1/3 \leq gene_{M+j} < 2/3 & x-z \text{ plane} \\ 2/3 \leq gene_{M+j} < 1 & y-z \text{ plane} \end{cases} \tag{6.1}$$

  By sorting the first M genes in ascending order, a BTPS is generated as well as a vector of Layer Filling Planes (LFP).

Figures 6.5 and 6.6 respectively illustrate the decoding procedure for the **strongly heterogeneous** and **weakly heterogeneous** chromosome representation. In the examples there are 4 types of boxes, b_1, b_2, b_3 and b_4, with $n_1 = 3$, $n_2 = 1$, $n_3 = 2$ and $n_4 = 2$.

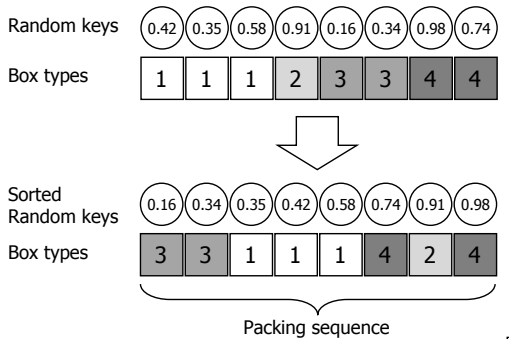


Figure 6.5: Strongly heterogeneous chromosome decoding procedure example

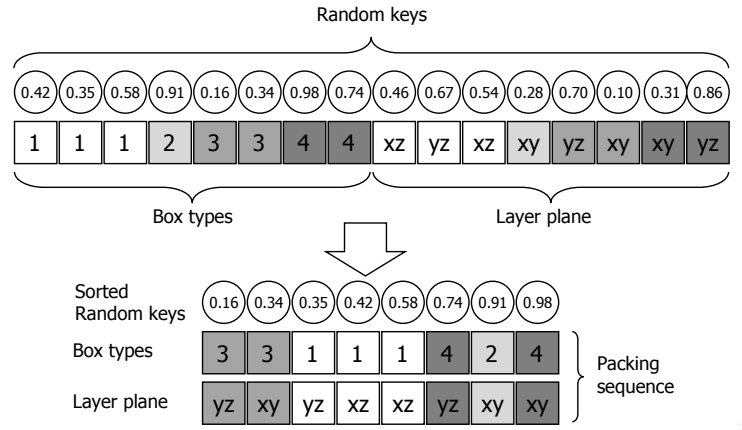


Figure 6.6: Weakly heterogeneous chromosome decoding procedure example

6.3.2 The placement heuristic

The placement heuristic is an iterative process with four main steps: selection of the box type, selection of the *maximal-space*, filling the *maximal-space*, and updating the system information. At each iteration the process tries to pack a layer of identical boxes in the container using a set of *maximal-spaces*. Three elements are combined in the process, a box type packaging sequence vector, a list of unpacked boxes and a list of *maximal-spaces*.

Step 0: Initialization

$S = \{C\}$, set of *maximal-spaces*
$B = \{b_1, b_2, ..., b_K\}$, set of box types—unpacked
$q_k = n_k$, number of boxes of type k—unpacked
$P = \varnothing$, set of boxes—packed

Step 1: Selecting the box type

The aim of the first step is to select the type of box to be loaded. The order in which the type of boxes are loaded into the container is provided by the BTPS vector obtained from the decoding procedure. The i^{th} type of box to be loaded is given by $BTPS(i)$. Successive iterations check to see if $q_k = 0$ for the selected box type k, whereby, if true, the next box type in the sequence is selected.

Step 2: Selecting the *maximal-space*

To select an empty *maximal-space* we use the back-bottom rule. The back-bottom rule first selects the *maximal-space* that is closest to the back of the container (x-axis), then selecting the space closest to the bottom of the container (z-axis), finally selecting whichever space is closer to one of the two back-bottom corners of the container (y-axis). The smallest distance between *maximal-space* s, with coordinates (x_{1s}, y_{1s}, z_{1s}) and (x_{2s}, y_{2s}, z_{2s}), to the two corners of the container, with coordinates $(0, 0, 0)$ and $(0, W, 0)$ respectively, is given by $y_{Vs} = min\{y_{s1}, (W - y_{s2})\}$.

Determining the smallest distance to the back-bottom corners of the container also determines the *maximal-space* Insertion Vertex, V, since it is the closest vertex to one of two back-bottom corners of the container (Figure 6.7). The Insertion Vertex determines the vertex of the *maximal space* where the build layer (see step 3) will be placed. Figure 6.8 illustrates two potential placements of a layer in a *maximal-space*.

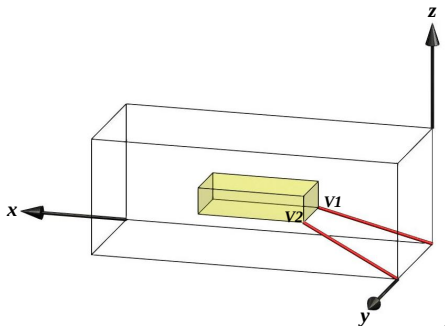


Figure 6.7: Distances between container corners and the *maximal-space* reference vertex

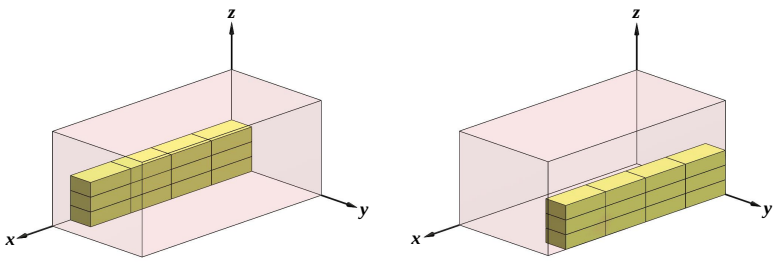


Figure 6.8: Two potential placements of a layer in a *maximal-space*

- The **back-bottom** rule selects from S the *maximal-space* s with the smallest x_s coordinate, then choosing that with the smallest z_s coordinate and finally that with the smallest y_{Vs}. Ties are broken by comparing the volume of the *maximal-spaces* and choosing the largest one.

This rule is similar to the back-bottom-left placement rule commonly used in CLP algorithms. However, by introducing a new potential insertion vertex, another surface of the container surface may be used, inducing the generation of larger empty spaces (Parreño et al., 2008).

Step 3: Filling the *maximal-space*

Having selected the box type and the *maximal-space*, the heuristic fills the *maximal-space* using box layers (that is, several identical boxes are arranged in rows and columns), constrained by the number of unpacked boxes. There are 6 possible layer arrangements for each box position (Figure 6.9). Considering that each box can be placed in up to 6 different positions (depending on the number of rotations allowed), a total of 36 layer types can be generated. When using the weakly heterogeneous indirect representation of a solution, this value is reduced to a maximum of 12, since the building plane is determined by the chromosome. In strongly heterogeneous problems, the number of boxes per type is smaller, resulting in layers of smaller dimension which reduces the impact of the layer building plane on the solution. The Best-fit criterion proposed by Parreño et al. (2008) was used to evaluate the layer configurations.

- The **Best-fit** criterion selects the layer with the best fit inside a *maximal-space*. The layer fit is determined by calculating the gap between the parallel faces of the layer and the *maximal-space*, ordering the values in non-decreasing order and choosing the best layer as given by the lexicographic order. Ties are broken using the number of inserted boxes criterion, (that is, the layer with the higher number of boxes).

If the selected box type does not fit the selected *maximal-space*, a new *maximal-space* is selected. If the selected box type does not fit any of the current *maximal-spaces* the box is skipped and the next box on the sequence is selected.

Step 4: Updating the system information

After packing a layer of m boxes of type k, the number of unpacked boxes q_k is updated. Also, the selected *maximal-space* s is removed from list S, new *maximal-spaces* are generated using the difference process and elimination process developed by Lai and Chan (1997). These processes not only generate new *maximal-spaces* from s, but also verify the intersection of the added layer with other *maximal-spaces* and eliminate *maximal-spaces* with dimensions that cannot be filled with the remaining unpacked boxes, thus saving computational time. The list of *maximal-spaces*, S, is then updated. The placement procedure is repeated until all the boxes are packed or there are no more *maximal-spaces* available.

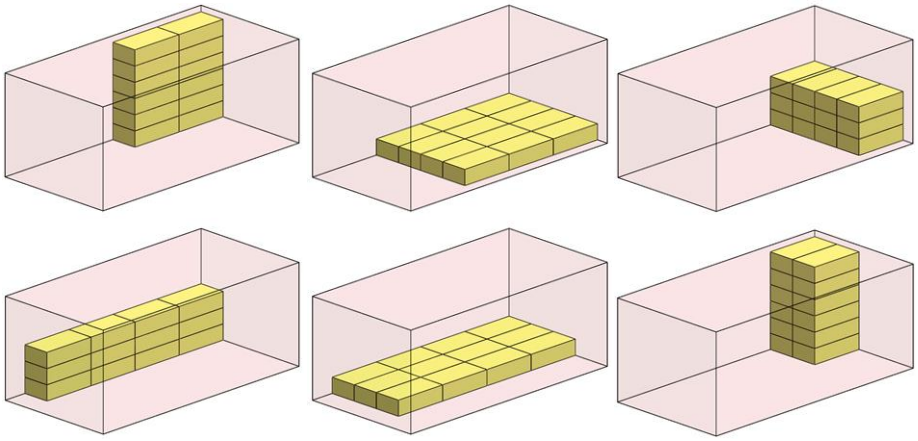


Figure 6.9: Example of the six different feasible layer types for a box position

6.3.3 Solution fitness computation

The constructive heuristic finishes by determining the percentage of the container volume packed using (6.2), where N_k is the number of loaded boxes of type k with volume $d_k \times w_k \times h_k$ packed in a container of volume $D \times W \times H$.

$$\frac{\sum_{k=1}^{K} N_k(d_k \times w_k \times h_k)}{D \times W \times H} \times 100\% \tag{6.2}$$

6.3.4 Static stability constraint

The static stability constraint is enforced by introducing a set of conditions during the filling of the *maximal-space* and the computation of the packed container volume, based on the static mechanical equilibrium conditions applied to rigid bodies that derive from Newton's first and third laws of motion.

The first condition of equilibrium is

$$\sum \overrightarrow{F} = \overrightarrow{0} \tag{6.3}$$

where $\overrightarrow{F}$ represents the external forces applied to a rigid body. A force is a vector quantity describing the force magnitude and the direction of its action (Hibbeler, 2010b).

The second condition of equilibrium is

$$\sum \overrightarrow{M_O} = \sum (\overrightarrow{r} \times \overrightarrow{F}) = \overrightarrow{0} \tag{6.4}$$

where $\overrightarrow{r}$ represents the vector from point O to the line of action of force $\overrightarrow{F}$. $\overrightarrow{M_O}$ represents the moment of a force, that is, the tendency of a force to rotate a body about a point or

axis. The first condition allows us to evaluate the translational equilibrium and the second condition allows us to evaluate the rotational equilibrium of a body (Hibbeler, 2010b).

These conditions, however, can only be fully validated with the complete loading arrangement. Therefore, in the proposed algorithm with static stability constraints, stability is first partially enforced in the *maximal-space* filling procedure, and secondly, in the fitness computation procedure, once a solution has been generated.

Filling the maximal-space

The partial static stability condition, enforced when filling the *maximal-spaces*, only evaluates the stability of a box in relation to its direct supporting boxes, when is placed in the selected *maximal-space*. As such, it considers that for every box, the resultant of the forces acting downwards, parallel to the z-axis (that is, the weight), is located on the geometric center of the box. Therefore, a box b can be loaded at (x, y, z), if the projection of its geometric center cg, with coordinates (x_{cg}, y_{cg}, z_{cg}) in plane $(0, 0, z)$, lies inside the box b support polygon SP. The support polygon SP of box b is formed by the convex hull of all horizontal support points of box b. The support polygon concept is frequently used in the research field of human movement simulation for stability modelling purposes (Badler et al., 1980; Vukobratović and Borovac, 2004). Therefore, to guarantee that the partial static stability condition during filling holds, one of the three following conditions must be satisfied.

1. the support polygon of the box is the container floor, that is, whenever z (the z-axis coordinate of a box) equals zero;
$$z = 0$$

2. the support polygon is defined by the top edges of a support box b_j (Figure 6.10); The condition holds for box b, supported by box b_j placed at $((x_{1,j}, y_{1,j}, z_{1,j}), (x_{2,j}, y_{2,j}, z_{2,j}))$, when
$$x_{1,j} \leq x_{cg} \leq x_{2,j}$$
$$y_{1,j} \leq y_{cg} \leq y_{2,j}$$
$$z = z_{2,j}$$

3. the support polygon is a convex polygon defined by the convex hull of the vertices of the polygons generated by the intersection of box b with its supporting boxes (Figure 6.11). Consider SP to be a convex polygon in plane $(0, 0, z)$ with m vertices $(p_0, p_1, ..., p_{m-1})$ defined by a sequence of points $(x_0, y_0), (x_1, y_1), ..., (x_{m-1}, y_{m-1})$ running in counterclockwise direction. Condition 3 holds whenever point cg is an interior point of the convex polygon SP. Point cg is an interior point of the convex polygon SP if cg is located on a line segment that connects a pair of points a, b in SP, such that $cg = \lambda a + (1 - \lambda)b$ for $0 \leq \lambda \leq 1$ (LaValle, 2006).

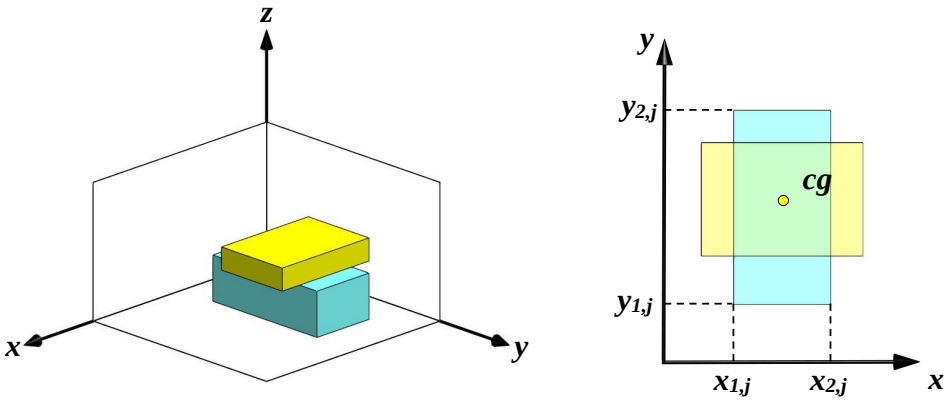


Figure 6.10: Condition 2 support polygon example

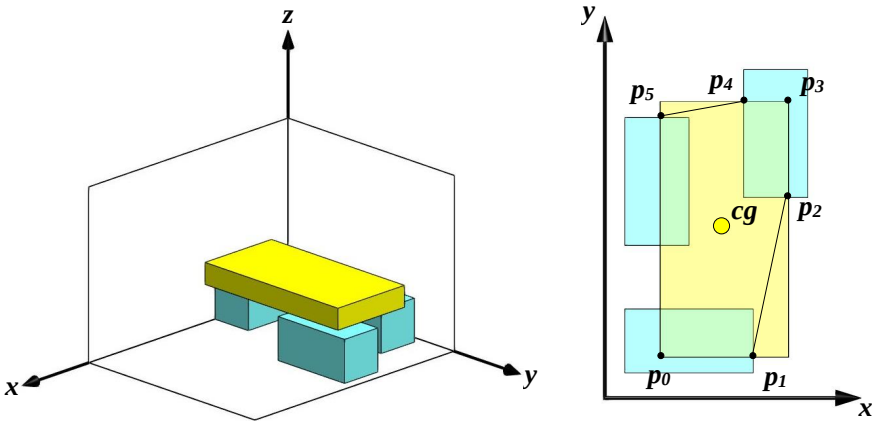


Figure 6.11: Condition 3 support polygon example

The algorithm for enforcing the stability condition during the filling of a *maximal-space* is described in Algorithm 5. A more detailed description of the *Gift wrapping* algorithm, that determines the support polygon of a box, and the *Point-in-Polygon* algorithm, that determines if a given point is in the interior of a convex polygon, is presented in Ramos et al. (2014b).

To the best of the authors' knowledge, there is no approach in the literature that combines the use of *maximal-spaces* and the evaluation of static stability, when there are gaps between supporting boxes, as proposed in this paper. Adding this possibility adds complexity to the management and filling of *maximal-spaces*, since it is required to determine the support boxes of the maximal-space in order to fill it and evaluate the fitness of the build layer configuration. However, this approach increases the flexibility of the algorithm.

The criterion used to evaluate layer configurations is the *best-overhanging* criterion.

- The **best-overhanging** criterion selects the layer which best fits the supporting boxes of the selected *maximal-space*. The layer fit is determined by calculating the gap

Algorithm 5 Static Stable Filling of a *Maximal-Space* (box b)

Input: Let box b be a box placed at coordinates $((x_1, y_1, z_1), (x_2, y_2, z_2))$
Output: Let *stable* be a boolean variable where TRUE represents a box that fills a *maximal-space* and FALSE otherwise

Begin
if $z = 0$ **then**
  return TRUE
end if
stable $\leftarrow$ FALSE
Let U be the set of boxes b_j that support box b
Let cg be the geometric centre of box b with coordinates x_{cg}, y_{cg}
for each box $b_i \in U$ **do**
  if $(x_{1,j} \leq x_{cg} \leq x_{2,j})$ and $(y_{1,j} \leq y_{cg} \leq y_{2,j})$ **then**
    stable $\leftarrow$ TRUE
  end if
end for
if *stable* = FALSE **then**
  for each box $b_j \in U$ **do**
    Determine the intersection vertices v with box b
  end for
  $SP \leftarrow$ **Call** Gift wrapping (v) ▷ Determine box b support polygon
  stable $\leftarrow$ **Call** Point-in-Polygon (cg, SP) ▷ Determine box stability
end if
return *stable*
End

between the parallel faces of the layer that are perpendicular to the $x-y$ plane and the support polygon of the selected *maximal-space*, ordering the values in non-decreasing order and choosing the best layer, given by the lexicographic order. Ties are broken by first evaluating the gap in the z-axis direction and secondly by calculating the additional number of boxes to be placed. Figure 6.12 illustrates the criterion.

Even thought the partial stability conditions by themselves do not guarantee the static stability of the cargo arrangement, when combined with the best-overhanging criterion, potentiate the generation of statically stable solutions.

Solution fitness computation

As in other approaches previously published in the literature, the metric used to evaluate the fitness of a solution is the percentage of the container volume packed by boxes. The difference in this algorithm is that only boxes that can be loaded in the container while

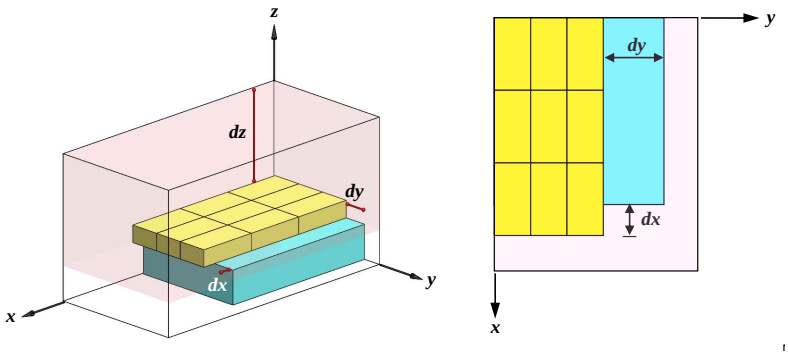


Figure 6.12: **Best-overhanging** criterion illustration example

guaranteeing static stability are considered, that is, boxes that would make the cargo unstable are withdraw from the physical packing sequence and their volume not considered.

Given that static stability conditions were already applied during the maximal-spaces filling procedure, how can cargo still be unstable? The conditions applied during the maximal-space filling procedure guarantee local stability, that is, that the arrangement inside the *maximal-space* is stable. However, the effect of filling that space on the remaining cargo is not taken into account because it would be unnecessarily restrictive. As the order by which *maximal-spaces* are filled during solution generation is not the the actual sequence by which boxes are loaded inside the container, when filling later on spaces physically related to the one that is currently being filled may revert a situation of potential instability into a globally stable cargo. For this reason, in the algorithm here proposed, static stability is locally enforced when filling the *maximal-spaces* and globally checked and imposed after the loading arrangement is finished, when the solution fitness is computed.

The final verification of the static stability of the finished loading arrangement is, therefore, done by the physical packing sequence algorithm (PPSA) with static stability as presented in Ramos et al. (2014b). The PPSA, given a container loading arrangement, generates a physical sequence by which each box can be actually loaded inside the container, considering static stability and efficiency of loading operations constraints. The approach to stability used in the PPSA is also based on the static mechanical equilibrium conditions applied to rigid bodies, derived from Newton's laws of motion. In generic terms, PPSA starts by determining a physical sequence by which boxes should be loaded in the container. Using this sequence, it loads one box at a time and for each one evaluates the static stability of the box and the static stability of the subset of all boxes already loaded. If, as a result of loading this new box, statical stability of the overall cargo is lost, then the box is removed from the physical packing sequence. As PPSA deals with a physical loading sequence and considers the effect of each box on all the previous loaded cargo, global static stability is guaranteed. As PPSA removes from the sequence (and from the solution) unstable boxes, the solution fitness (percentage of the container's space that is used) reflects only statically stable boxes.

6.4 Computational experiments

This section presents the results of the computational experiments run to evaluate the efficiency of the proposed container loading algorithm with static stability (CLA-SS). The proposed algorithm is run without any static stability constraint (Unsupported), with the classical full base support constraint (Full Support) and with the new static stability constraint (Supported), with the results then being compared among themselves. These results are also compared with the most recent and efficient approaches to the CLP. The algorithm was coded in Visual C++ and run on an computer with 2 Intel Xeon CPU E5-2687W at 3.1Ghz with 128 Gigabytes of RAM running the Windows 7 Pro 64 bit operating system.

Two versions of the algorithm are tested contrasting the use of strongly heterogeneous (CLA-SS(S)) and weakly heterogeneous (CLA-SS(W)) chromosome indirect representations. We start by comparing the CLA-SS(S) and CLA-SS(W) Unsupported and Full Support variants with the most efficient algorithms. Then we compare the CLA-SS Unsupported, Full Support and Supported variants, for both CLA-SS(W) and CLA-SS(W) versions. Finally, we compare the Supported variant of the two CLA-SS versions with the most efficient algorithms that guarantee full base support, and thus indirectly also guarantee static stability. The column headings for the computational results in the tables that follow refer to the different algorithms used, namely: BSG—the Beam Search Approach of Araya and Riff (2014); HBMLS—the block-loading heuristic based on multi-layer search of Zhang et al. (2012) (two versions are presented, the AS version that uses simple blocks and the AC version that uses composite bocks); ID-GLTS—the iterative-doubling greedy-lookahead algorithm of Zhu and Lim (2012); BRKGA—the multi-population biased random-key genetic algorithm of Gonçalves and Resende (2012); and CLA-SS—the proposed container loading algorithm with static stability.

6.4.1 Test problem instances

The problem tests used to evaluate the effectiveness of the new algorithm are the 1500 problems proposed by Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999). These instances are organized in 15 classes, with a total of 100 instances per class. They are designated here as BR1 to BR15. The instances used cover a wide range of situations. The heterogeneity of the boxes increases from just 3 different box types in BR1 to 100 box types in BR15. The number of boxes per box type also varies from 50.15 boxes per type in BR1 to 1.33 in BR15. The dimensions of the boxes are generated independently from the dimensions of the container and the total volume of the boxes in each individual instance never exceeds the container volume. On average the total volume of the boxes to be packed represents 99.46% of the container volume. All tables presenting computational results show the average solution value, in terms of percentage of container space utilization, for the 100 instances of each class, the aggregate results for the class instances BR1–7 and BR8–15, and the aggregate results for the class instances BR1–15.

6.4.2 Genetic algorithm parameters

The genetic algorithm parameters used in the algorithm are based on the recommended parameter value settings proposed by Gonçalves and Resende (2011) for generic BRKGA, and the parameters used by Gonçalves and Resende (2012) and Gonçalves and Resende (2013) for BRKGA developed for the CLP. Preliminary computational experiments allowed the selection of the parameters presented in Table 6.2.

Table 6.2: Genetic algorithm parameters used in all computational experiments

Parameters	Values
Top	15%
Bottom	15%
Crossover probability	0.7
Population size	20 × number of boxes
Number of populations	2
Exchange between pop.	Every 15 generations
Fitness function	Maximize the % of packed container volume
Stopping criteria	after 1000 generations

6.4.3 Static Stability Algorithm performance compared to other CLP algorithms

The first set of experiments aimed to compare the two CLA-SS versions against the most efficient algorithms, that can be found in the literature, using both Unsupported and Full Support variants. Comparing the results (presented in Table 6.3) of the Unsupported variant against the equivalent algorithms (BSG, HBMLS(AS), HBMLS(AC) and ID-GLTS) there is an overall difference to the best (BSG) of -0.98 percentage points, for the CLA-SS(S), and -0.69 for the CLA-SS(W). It must be stated that the BSG outperforms all the algorithms in the Unsupported variant. A comparison against the original BRKGA algorithm shows an overall difference of -0.12 percentage points to the CLA-SS(S) and of 0.17 to the CLA-SS(W). It can be observed that both CLA-SS versions underperform against the BRKGA for weakly heterogeneous instances, and outperform for the strongly heterogeneous ones.

Analysing the results for the Full Support variant (presented in Table 6.4) show that the equivalent algorithms with the best performance, BSG and HBMLS(AC), have an overall difference to the CLA-SS(S) of -0.53 and -0.29 percentage points respectively and an overall difference to the CLA-SS(W) of -0.05 and 0.19 percentage points; comparing with the results for the original BRKGA algorithm there is an overall difference of 0.10 percentage points to the CLA-SS(S) and 0.58 to the CLA-SS(W). Notably, the CLA-SS(W) presents the best overall results for the BR8–15 classes of problems.

These results show that the proposed versions of the CLA-SS algorithm have a level of performance similar to the best-in-class CLP algorithms, both in the Unsupported and

Table 6.3: Performance comparison of Unsupported variants

Class Problems	BSG (2014)	HBMLS (AS) (2012)	HBMLS (AC) (2012)	ID-GLTS (2012)	BRKGA (2010)	CLA-SS(S)	CLA-SS(W)
BR_1	**95,69**	94,87	93,54	95,59	95,28	93,54	95,10
BR_2	**96,24**	95,41	94,47	96,13	95,90	94,83	95,76
BR_3	**96,49**	95,56	95,12	96,30	96,13	95,40	95,85
BR_4	**96,31**	95,38	95,10	96,15	96,01	95,38	95,74
BR_5	**96,18**	95,22	95,08	95,98	95,84	95,43	95,65
BR_6	**96,05**	95,10	95,21	95,81	95,72	95,36	95,61
BR_7	**95,77**	94,69	94,87	95,36	95,29	95,18	95,32
BR_8	**95,33**	94,16	94,60	94,80	94,76	94,80	95,07
BR_9	**95,07**	93,76	94,24	94,53	94,34	94,64	94,77
BR_10	**94,97**	93,38	94,08	94,35	93,86	94,32	94,47
BR_11	**94,80**	92,87	93,86	94,14	93,60	94,01	94,14
BR_12	**94,64**	92,59	93,67	94,10	93,22	93,77	93,93
BR_13	**94,59**	92,25	93,45	93,86	92,99	93,56	93,37
BR_14	**94,49**	91,84	93,34	93,83	92,68	93,28	93,22
BR_15	**94,37**	91,53	93,14	93,78	92,46	92,81	92,65
Mean (BR 1-7)	**96,11**	95,18	94,77	95,90	95,74	95,02	95,58
Mean (BR 8-15)	**94,78**	92,80	93,80	94,17	93,49	93,90	93,95
Mean (BR 1-15)	**95,40**	93,91	94,25	94,98	94,54	94,42	94,71

∗ **The best values appear in bold**

Full Support variants.

The comparison of computational times between the different approaches would only provide significant information if the software programming technologies, such as the use of parallel programming or memory management, were similar, and the tests run on identical computers and operating systems. Regardless this fact, Table 6.5 presents the reported average running time of each approach.

6.4.4 Container Loading Algorithm with Static Stability performance for the different versions

A second set of experiments evaluates the performance of the two CLA-SS versions(CLA-SS(S) and CLA-SS(W)) across the Unsupported, Supported and Full Support variants. The average results obtained are presented in Table 6.6. In the case of the Unsupported variant, the results show that the CLA-SS(S) performs better over the classes BR13 to BR15; in the case of the Supported variant the CLA-SS(S) performs better over the classes BR9 to BR15; and in the case of the Full Support variant, the CLA-SS(W) always returns the best performance. It can also be observed that for each class, the average Unsupported results are higher than the comparable results of the Supported variant, which in turn are higher than the respective Full Support results.

Table 6.7 presents the number of times for each class and variant, that each of the CLA-SS versions outperforms the other. All three CLA-SS(W) variants outperform the CLA-SS(S) variants in the weakly heterogeneous classes of instances. As for the strongly heterogeneous classes, CLA-SS(S) outperforms CLA-SS(W) in the Supported variant. However

Table 6.4: Performance comparison of Full Support variants

Class Problems	BSG (2014)	HBMLS (AS) (2012)	HBMLS (AC) (2012)	ID-GLTS (2012)	BRKGA (2010)	CLA-SS(S)	CLA-SS(W)
BR_1	**94,50**	94,30	93,95	94,40	94,34	92,36	93,86
BR_2	**95,03**	94,74	94,39	94,85	94,88	93,68	94,55
BR_3	**95,17**	94,89	94,67	95,10	95,05	94,21	94,75
BR_4	**94,97**	94,69	94,54	94,81	94,75	94,23	94,63
BR_5	**94,80**	94,53	94,41	94,52	94,58	94,05	94,38
BR_6	**94,65**	94,32	94,25	94,33	94,39	93,87	94,24
BR_7	**94,09**	93,78	93,69	93,59	93,74	93,26	93,82
BR_8	93,15	92,88	93,13	92,65	92,65	92,64	**93,16**
BR_9	92,53	92,07	92,54	92,11	91,90	92,13	**92,62**
BR_10	92,04	91,28	92,02	91,60	91,28	91,62	**92,09**
BR_11	91,40	90,48	91,45	90,64	90,39	91,19	**91,56**
BR_12	90,92	89,65	90,91	90,35	89,81	90,91	**91,28**
BR_13	90,51	88,75	90,43	89,69	89,27	90,66	**90,93**
BR_14	89,93	87,81	89,80	89,07	88,57	90,31	**90,38**
BR_15	89,33	86,94	89,24	88,36	87,96	90,01	**90,08**
Mean (BR 1-7)	**94,74**	94,46	94,27	94,51	94,53	93,66	94,32
Mean (BR 8-15)	91,22	89,98	91,19	90,56	90,23	91,18	**91,51**
Mean (BR 1-15)	**92,87**	92,07	92,63	92,40	92,24	92,34	92,82

∗ **The best values appear in bold**

Table 6.5: Average computational times (s) for test classes BR1 to BR15

	BSG	HBMLS	ID-GLTS	BRKGA	CLA-SS(W)
Unsupported	500	597	500	147	274
Full Support	150	519	150	232	146

Table 6.6: Performance comparison of CLA-SS versions

Class	Unsupported		Supported		Full Support	
Problems	CLA-SS(S)	CLA-SS(W)	CLA-SS(S)	CLA-SS(W)	CLA-SS(S)	CLA-SS(W)
BR_1	93,54	**95,10**	93,16	**94,79**	92,36	**93,86**
BR_2	94,83	**95,76**	94,73	**95,40**	93,68	**94,55**
BR_3	95,40	**95,85**	95,21	**95,55**	94,21	**94,75**
BR_4	95,38	**95,74**	95,29	**95,51**	94,23	**94,63**
BR_5	95,43	**95,65**	95,15	**95,43**	94,05	**94,38**
BR_6	95,36	**95,61**	95,09	**95,31**	93,87	**94,24**
BR_7	95,18	**95,32**	94,93	**95,14**	93,26	**93,82**
BR_8	94,80	**95,07**	94,69	**94,78**	92,64	**93,16**
BR_9	94,64	**94,77**	**94,51**	94,45	92,13	**92,62**
BR_10	94,32	**94,47**	**94,07**	93,95	91,62	**92,09**
BR_11	94,01	**94,14**	**93,68**	93,38	91,19	**91,56**
BR_12	93,77	**93,93**	**93,23**	92,61	90,91	**91,28**
BR_13	**93,56**	93,37	**92,59**	91,64	90,66	**90,93**
BR_14	**93,28**	93,22	**91,68**	90,72	90,31	**90,38**
BR_15	**92,81**	92,65	**90,58**	90,10	90,01	**90,08**
Mean (BR 1-7)	95,02	**95,58**	94,79	**95,30**	93,66	**94,32**
Mean (BR 8-15)	93,90	**93,95**	**93,13**	92,70	91,18	**91,51**
Mean (BR 1-15)	94,42	**94,71**	93,91	**93,92**	92,34	**92,82**

∗ **The best values of each variant (that is, Unsupported, Supported and Full Support) appear in bold**

in the Unsupported and Full Support variants it is CLA-SS(W) that has the best overall performance.

Table 6.7: Number of times the CLA-SS versions provided better solutions

Class	Unsupported		Supported		Full Support	
Problems	CLA-SS(S)	CLA-SS(W)	CLA-SS(S)	CLA-SS(W)	CLA-SS(S)	CLA-SS(W)
BR_1	2	96	4	94	5	86
BR_2	5	93	18	81	13	85
BR_3	22	78	32	68	18	82
BR_4	25	75	35	65	31	68
BR_5	32	68	32	68	32	68
BR_6	28	70	33	65	31	69
BR_7	40	60	33	65	28	72
BR_8	25	72	45	55	22	78
BR_9	41	59	52	48	22	78
BR_10	44	56	55	45	23	76
BR_11	42	56	53	47	33	67
BR_12	30	70	63	37	27	73
BR_13	53	46	71	29	35	63
BR_14	52	48	69	31	49	51
BR_15	59	41	59	41	46	53
Mean (BR 1-7)	22.0	77.1	26.7	72.3	22.6	75.7
Mean (BR 8-15)	43.3	56.0	58.4	41.6	32.1	67.4
Mean (BR 1-15)	33.3	65.9	43.6	55.9	27.7	71.3

The impact of the number of generations of the algorithm is depicted in Figure 6.13. The evolution of the solution is fastest under the Full Support variant, with the Unsupported and the Supported variants requiring progressively more iterations.

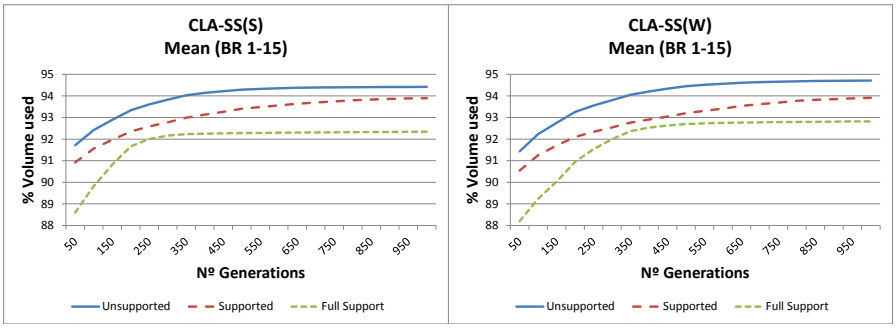


Figure 6.13: Influence of the number of generations

6.4.5 Performance of statically stable algorithms

Finally, the CLP algorithms with the best performance in the Full Support variant (BSG and CLA-SS(W)), that is, with static stability, are compared with the two versions of the CLA-SS Supported variant. The average results obtained are presented in Table 6.8. The Supported variant CLA-SS(W), had the best average performance for BR1 to BR8 classes

of instances, while the CLA-SS(S) version outperformed all approaches for BR9 to BR15 classes of instances.

Table 6.8: Performance comparison of statically stable solutions

Class Problems	BSG (2014)	CLA-SS(W) Full Support	CLA-SS(S) Supported	CLA-SS(W) Supported
BR_1	94,50	93,86	93,16	**94,79**
BR_2	95,03	94,55	94,73	**95,40**
BR_3	95,17	94,75	95,21	**95,55**
BR_4	94,97	94,63	95,29	**95,51**
BR_5	94,80	94,38	95,15	**95,43**
BR_6	94,65	94,24	95,09	**95,31**
BR_7	94,09	93,82	94,93	**95,14**
BR_8	93,15	93,16	94,69	**94,78**
BR_9	92,53	92,62	**94,51**	94,45
BR_10	92,04	92,09	**94,07**	93,95
BR_11	91,40	91,56	**93,68**	93,38
BR_12	90,92	91,28	**93,23**	92,61
BR_13	90,51	90,93	**92,59**	91,64
BR_14	89,93	90,38	**91,68**	90,72
BR_15	89,33	90,08	**90,58**	90,10
Mean (BR 1-7)	94,74	94,32	94,79	**95,30**
Mean (BR 8-15)	91,22	91,51	**93,13**	92,70
Mean (BR 1-15)	92,87	92,82	93,91	**93,92**

∗ **The best values appear in bold**

6.5 Conclusion

In this paper we addressed the static stability constraint within the three-dimensional rectangular single CLP. We proposed two versions of an hybrid genetic algorithm based on a multi-population biased random key genetic algorithm and a constructive heuristic that uses a two dimensional box arrangement strategy and a *maximal-spaces* representation of empty spaces inside the container. A new procedure for filling the *maximal-spaces*, based on the static mechanical equilibrium conditions applied to rigid bodies derived from Newton's laws of motion, that allows the evaluation of stability when there are gaps between supporting boxes, was also proposed. The two versions of the algorithm were tested using the well known benchmark instances of Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999) and compared to the best known CLP solutions in the literature. The proposed approach improved the best known average results for the instances BR-8 to BR15 from 91.22% to 91.51%, using the full base support constraint, and improved the overall average of the solutions from 92.87% to 93.92%, for statically stable solutions.

Chapter 7

Cargo dynamic stability in the container loading problem – A physics simulation tool approach

A. Galrão Ramos[†] [‡], João Jacob[†], José F. Oliveira[†]
Jorge F. Justo[‡], Rui Rodrigues [†] and Miguel Gomes[†]

[†] INESC-TEC and Faculty of Engineering, University of Porto
[‡] CIDEM, School of Engineering, Polytechnic of Porto

Abstract

The container loading problem (CLP) is a real-world driven, combinatorial optimization problem that addresses the maximization of space usage in cargo transport units. The research conducted on this problem failed to fulfill the real needs of the transportation industry, due to the inadequate representation of practical-relevant constraints. The dynamic stability of cargo is one of the most important practical constraints. It has been addressed in the literature in an over-simplified way which does not actuality translate real-world stability. This paper proposes a physics simulation tool based on a physics engine, which can be used to translate real-world stability into the CLP. To validate the tool, a set of benchmark tests is proposed and the results obtained with the physics simulation tool are compared with the state-of-the-art simulation engineering software Abaqus Unified FEA. Analytical calculations have been also conducted, and it was also possible to conclude that the tool proposed is a valid alternative.

7.1 Introduction

Using transportation resources efficiently is very relevant in the field of logistics, and has an impact on operational efficiency, customer satisfaction, and transport safety. The Container Loading Problem (CLP) addresses the optimization of the spatial arrangement of cargo inside containers so that the utilization of the space is maximized. The problem belongs to the wider combinatorial optimization class of Cutting and Packing problems. According to the typology for cutting and packing problems proposed by Wäscher et al. (2007), problems can be classified according to dimensionality, assortment of large items, assortment of small items, assignment type and shape of small items. This paper will focus on three-dimensional rectangular placement problems. The CLP can have two main variants: the maximization of the value of the cargo loaded when the number of containers is not sufficient to accommodate the entire cargo, or the minimization of the value of containers when there are sufficient containers to accommodate the entire cargo.

In order to be used in real world scenarios, it is necessary to consider a number of constraints found in practice when addressing the CLP. Cargo stability, weight distribution, cargo positioning or cargo orientation constraints are just some examples (Bortfeldt and Wäscher, 2013).

Stability is considered one of the most important CLP constraints and has received plenty of attention from a large number of authors (Bortfeldt and Wäscher, 2013). Existing approaches to stability can be classified in two main groups, one that only addresses static stability, and another that addresses static and dynamic stability. Static stability refers to the ability of each box to maintain the loading position during loading operations, and dynamic stability refers to the ability of each box to maintain the loading position during transportation.

Dynamic stability is usually ensured by placing the boxes with their sides adjacent to other boxes or container walls. The metric used to evaluate dynamic stability is usually the lack of lateral support, that is, the percentage of boxes that do not have at least three of their four lateral sides in contact with other boxes or with the container walls (Bortfeldt and Wäscher, 2013). This approach is used as a proxy of the real-world dynamic stability constraint and has been conditioning the algorithms developed for this problem. However, its effectiveness as a dynamic stability constraint can be easily dismissed. In a wall of boxes, as illustrated in Figure 7.1, boxes can have 3 sides of lateral support, but if there is an acceleration along the x-axis the boxes will most likely fall.

This is a simple example of how the existing approaches do not take into account a series of real-time dynamic interactions and constraints, even thought they can be easily incorporated into the CLP algorithms without being computationally expensive.

Dynamic stability is affected by the vehicle's acceleration, which depends on the path, pavement conditions, traffic conditions, and ultimately the driver's behaviour, among other factors. Furthermore, the mechanical coupling between the wheels, suspension, vehicle structure and container platform also influence the forces that are ultimately applied to the cargo.

Nevertheless, at a fine-grained abstraction level, all these factors can be represented as

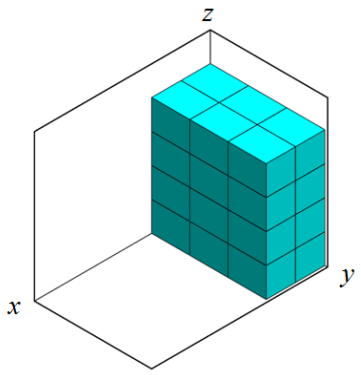


Figure 7.1: Unstable pattern example

a set of time-varying forces that are applied to the container and transferred to the cargo itself (which are then affected by the interactions between the boxes).

Therefore, it is possible to divide the problem of analyzing dynamic cargo stability into two parts:

(a) calculating the forces resulting from the multiple factors aforementioned, and;

(b) simulating the result of applying those forces to the cargo container.

This paper focuses on part (b). Assuming that there is a description of a series of time-varying forces to be applied to the container and cargo arrangement, the goal is to have a tool that simulates and graphically renders, in real time, the result of applying those forces to the cargo. This makes it possible not only to visualize and record the effects of those forces on the individual boxes that compose the cargo (such as sliding, rotating, tipping, falling), but also to extract dynamic stability metrics. Part (a) is outside the scope of this paper.

The simulation of Newtonian physics, including rigid body dynamics, is implemented in software components known as physics engines. These engines can be commonly found in the context of game development, as they are inherently designed to simulate various physical phenomena in real time. Therefore, the goal here was to use a real-time physics engine as the basis for the dynamic stability real-time simulation tool.

Section 7.2 presents a review of state-of-the-art physics engines and their features in terms of quality and performance of the simulation, to assess if they are suitable in this context.

Section 7.3 describes the simulation tool we developed – StableCargo –, detailing the underlying architecture and functioning.

Different tests were conducted to assess the usefulness and quality of the simulation, comparing the simulation generated by StableCargo with the results obtained with a state-of-the-art engineering simulation software (Abaqus Unified FEA) and analytical calculation. These tests and results are presented in Section 7.4.

Finally, conclusions are drawn in Section 7.5, along with possible future research topics in this area.

7.2 Related work

A physics engine is a computer software designed to simulate various physical phenomena, such as rigid body dynamics, soft body dynamics or fluid dynamics. It manages the forces applied to objects and the interactions between objects by simulating Newtonian physics (Jones, 2011; Seugling and Rolin, 2006). According to Erleben (2002), a physics engine has two main components: collision detection and dynamic simulation. Each one consists of a set of four interacting modules (see Figure 7.2).

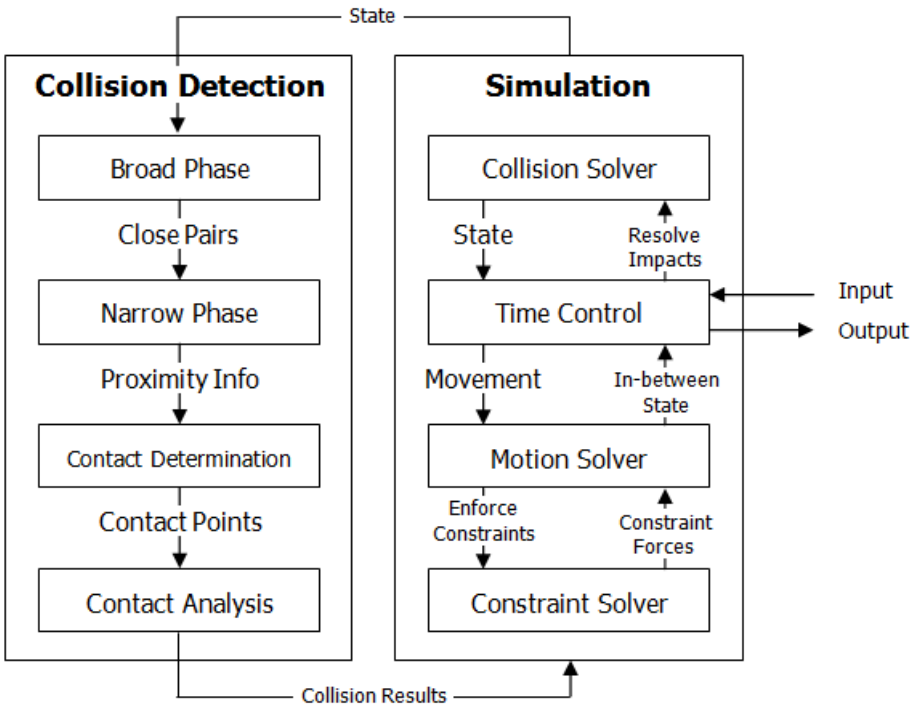


Figure 7.2: General Purpose Module design (Erleben, 2002)

Their performance is influenced by six essential factors: the simulator paradigm, the integrator, the object representation, the collision detection and contact determination, the properties of the material and the constraint implementation (Boeing and Bräunl, 2007). These factors are usually defined to address a specific application (Boeing and Bräunl, 2007).

Physics engines are used in video-games and simulations in order to mimic real world physics. Pepper et al. (2007) developed a methodology for testing the validity of robot models that can be done either physically or through a simulator that is based on a physics engine. This way, it is possible to develop and test new robot models without actually testing them in the real world, thus potentially reducing costs. Examples include Microsoft's Flight Simulator and X-Plane approved by the FAA (Federal Aviation Administration), meaning that flight hours logged in the simulator count for certification purposes, although there are some limits (Williams, 2011). These are also often used for training purposes

(Koonce and Bramble Jr., 1998). Another possible use is in MAS (Multi Agent Systems) where physical restrictions are to be enforced. Multi-agent simulations (Candea et al., 2001; Pinciroli et al., 2011) or crowd behavior simulations (Braun et al., 2003; Almeida et al., 2014) are useful for simulating disaster scenarios or search and rescue operations, benefiting from the usage of physics engines.

The evaluation or validation of physics engines has been addressed by various authors. Seugling and Rolin (2006) and Boeing and Bräunl (2007) evaluated physics engines generally, without focusing on a particular application, while the evaluation by Hummel et al. (2012) focused on an interactive application for on-orbit servicing tasks. Pepper et al. (2007) focused on determining and increasing simulation accuracy in urban search and rescue (USAR) robot simulation. The physics engines evaluated or validated in each paper are presented in Table 7.1.

Table 7.1: Physics engines validated in the literature

	(Seugling and Rolin, 2006)	(Boeing and Bräunl, 2007)	(Pepper et al., 2007)	(Hummel et al., 2012)
Open Dynamics Engine	x	x		x
PhysX	x	x		x
Newton Game Dynamics	x	x		x
Tokamak		x		
True Axis		x		
Bullet Physics		x		x
JigLib		x		
Unreal Engine 2.0			x	
Havok Physics				x

The physics engines were evaluated or validated by performing and measuring a set of tests. Seugling and Rolin (2006) developed nine tests in order to evaluate three features: energy preservation, constraint handling and collision detection. Boeing and Bräunl (2007) tested the integrator's performance, the properties of the material, the constraint stability, the collision system and the object stacking. Hummel et al. (2012) focused on collision detection, accuracy of collision, constraint stability and collision and friction of complex geometric objects. Pepper et al. (2007) used a set of tests from the National Institute of Standards and Technology (NIST) standard test methods for USAR robots to compare reality and virtual simulation.

From the benchmark tests performed, Seugling and Rolin (2006) reported that Newton Game Dynamics (Jerez and Suero, 2003) had the best overall results, while Boeing and Bräunl (2007) stated that Bullet Physics (Coumans, 2013) had the best overall performance. Finally, Hummel et al. (2012) considered that Newton Game Dynamics and PhysX (Corporation, 2004) can compete with Bullet Physics.

The use of simulation tools based on physics engines to analyse the stability of cargo during transportation within the CLP was first presented in the work of Mustafee and Bischoff (2013). However, the authors use it as a proof-of-concept for modelling scenarios in agent based simulation, only performing some exploratory experiences.

7.3 StableCargo – physics simulation tool

The work presented in this paper focuses on a tool – StableCargo – developed to simulate the dynamic forces applied to the cargo in a container, as well as on the visualization and extraction of metrics to assist in the assessment of dynamic stability. The goal is to improve the development of spatial optimization algorithms that are responsible for creating cargo arrangements. This way, it is possible to test if a given cargo arrangement is stable under a given scenario (that is, a given set or sequence of accelerations) or not.

Therefore, this tool can be integrated with other tools in a more complex workflow, including not only tools which can, for example, generate initial statically-stable cargo arrangements, for describing the dynamic transportation conditions (such as vehicle path, weather conditions, traffic), but also post-simulation analysis and evaluation tools.

With this in mind, the following assumptions and requirements were considered when designing the tool:

- The inputs provided include one or more pre-generated cargo arrangements and a time-varying series of accelerations;
- The user can also provide box type files that specify the density and friction coefficient for each type of box present in the respective cargo arrangement;
- The visualization will make it possible to visualize an interactive 3D representation of the cargo, including virtual camera motion to allow a detailed inspection of particular cargo areas;
- The forces will be applied to each of the cargo arrangements according to the time sequence, and rigid body dynamics will be computed and visualized in real time in the interactive 3D view;
- The tool can be executed in batch mode, without visualization, and if possible simulating at speeds superior to real time, allowing faster simulations of multiple paths and cargo arrangements;
- The tool should be multi-platform as much as possible, so that it can be integrated with tools in different operating systems;

- The tool should output both raw information regarding the motion sustained by the cargo elements, as well as a set of metrics computed taking into consideration the raw data (for example indicate how unstable the cargo was overall, the maximum displacement or rotation sustained by any box).

To support the multi-platform and real time 3D requirements, and taking into consideration the review of the state-of-the-art, the technologies chosen to support the tool were the CGFLib library (Rodrigues et al., 2012) and the Bullet physics engine (Coumans, 2013). Both of these libraries are written in C++ language, are cross-platform and can run on most hardware used today.

The CGFLib library is a computer graphics library built over OpenGL, GLUT (Shreiner and The Khronos OpenGL ARB Working Group, 2009) and GLUI (Rademacher, 2006) that has been developed by part of the team as a wrapper for rapid 3D cross-platform application development using those technologies. In StableCargo it is used as the basis for the 3D rendering and graphical user interface. The Bullet physics engine was chosen for the physics simulation over the PhysX and Newtons Dynamics engines due to its support of OpenCL, which makes it possible to improve computational speed in parallel CPU or GPU architectures (which, despite not being explored at the moment, is one of the goals for the future), and the availability of technical documentation and community support. Figure 7.3 presents the tool's workflow.

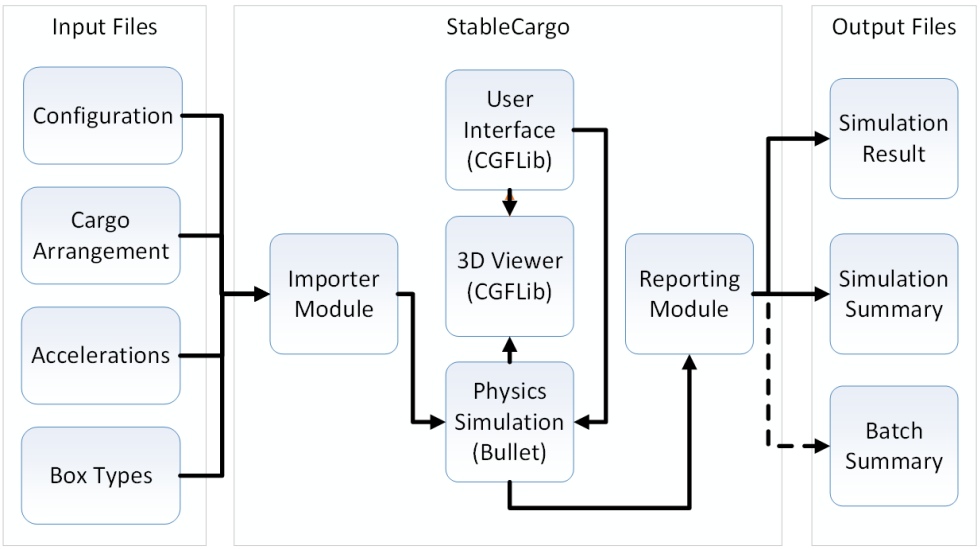


Figure 7.3: StableCargo's workflow

To evaluate the dynamic stability of a cargo arrangement, a set of metrics is proposed that reflects the damage suffered by a box during transportation. Damage is considered to exist if a box falls or there is a combination of velocity change and an acceleration peak that cause damage to the box. This concept was introduced in the field of packaging design by Newton (1968) to evaluate product fragility and it is known as the Damage Boundary

Curve (DBC). The DBC divides the plane, with the velocity change as the horizontal axis and the acceleration peak as the vertical axis, in two regions (see Figure 7.4), the Damage Region and the No Damage Region (Newton, 1968). The DBC is usually determined experimentally in laboratory tests.

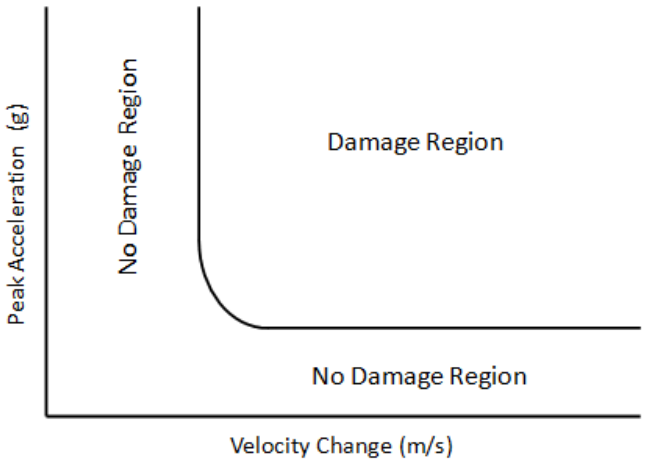


Figure 7.4: Typical Damage Boundary Curve

The following subsections describe the simulation and the viewing tool (including a description and images of the interface), the type of information and format of the input files that provide cargo arrangements, accelerations and other configurations for the simulator to run, and the output information provided to the file as a result of the simulations.

7.3.1 The StableCargo application and interface

The StableCargo physics simulator serves both as an interactive 3D visualizer of dynamic cargo simulations, or a batch simulator. When used in an interactive mode, the user can load cargo arrangements and apply to them a set of time-varying accelerations from a file, and thus visualize a 3D representation of the simulation.

When loaded in batch mode, StableCargo can perform multiple simulations in sequence, and output the results to the file. These modes and their operation details are controlled by the input files, described in the following subsection.

The application window is divided into two areas: the 3D view and the GUI (see Figure 7.5).

In the 3D view, it is possible to see a transparent (wireframe) representation of the container at all times. The view point around the container can be changed and zoomed using the mouse at any time (including during simulations), and the container is kept within the frame. It is also possible to see the timed elapsed since the beginning of a simulation, in s, as well as the linear speed of the container, in km/h.

If enabled on the *Configuration* file, the user can also see an overlay texture on the ground that marks every 50 m. This allows the user to better perceive the scale of the

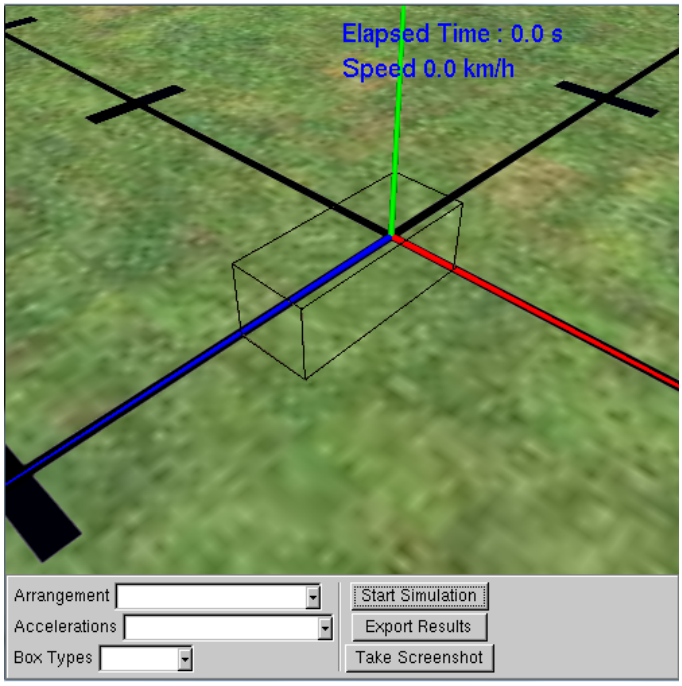


Figure 7.5: The StableCargo physics simulator

simulation and its elements (speed, translation of the container, size of container and boxes).

When cargo is present, it is represented as boxes of varying visual aspect, to make it easier to differentiate types of boxes and orientations (see Figure 7.6). Each box type is represented by a unique box image (texture), and the faces are numbered by dots (similar to dice) to identify the sides of the box, and thus show its orientation. The textures used for boxes and markings are configurable, and they are blended in real time using an OpenGL Shading Language shader.

The GUI area contains a set of widgets that are used for controlling the application interactively. The controls "Arrangement", "Accelerations" and "Box Types" allow the user to select a set of *Cargo Arrangement*, *Accelerations* and *Box Types* files that will be used for a simulation. The button "Start Simulation" will load the selected *Cargo Arrangement*, *Acceleration* and *Box Types* files, setup the simulation and run it.

Loading takes place via the *Importer Module*, as depicted in Figure 7.3. After loading the files, each box described in the cargo arrangement is instantiated in the physics engine as a single rigid body (undeformable physical entity) with mass or density as specified by its type and the information in the *Configuration* file. The centre of mass is considered as being at the centroid of the respective box. The container itself is also created as a physical entity within the physics engine. It consists of six rigid bodies (the walls, floor and ceiling), comprising a compound entity that represents a hollow parallelogram with the dimensions of a standard 20 feet container with a mass of 3700 kg, and a friction coefficient as specified in the *Configuration* file. In Bullet, friction is declared per physical entity.

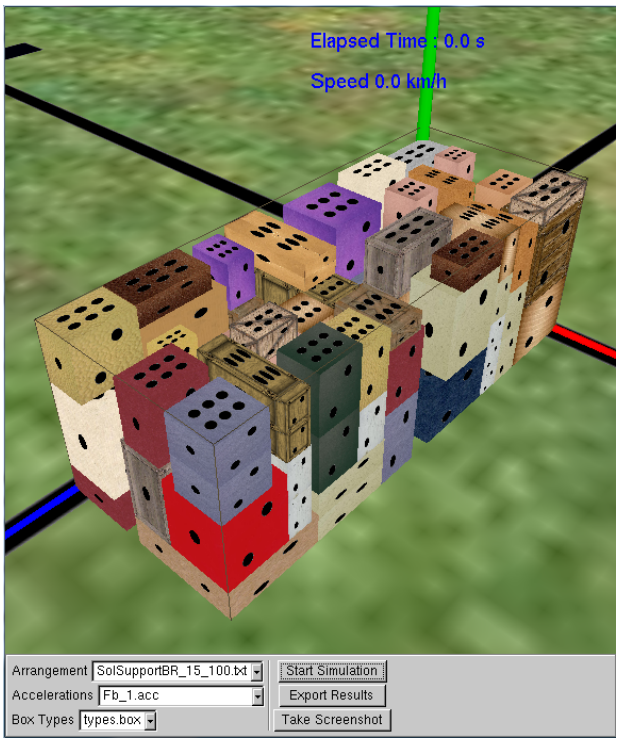


Figure 7.6: Representation of a loaded *Cargo Arrangement* file

When two objects are colliding, the friction force is obtained by multiplying the friction coefficients of both objects. This translates to a model compatible with the Newtonian model of a friction coefficient between pairs of objects. Parametrisation of the size and mass of a container were considered unnecessary for the scope of this project, as the CPL problem currently under study considers only this type of container.

With the physics setup completed, the cargo is rendered in the 3D view (Figure 7.6), and the simulation starts. Forces are calculated and applied to all objects, and these will move accordingly until the simulation is concluded. The user can change the viewpoint at any time using the mouse.

During the simulation, the *Reporting Module* generates internal statistics concerning each box (such as, if the box has fallen). The current results can be exported on demand via the "Export Results" button, either during the simulation (if the user is interested in analyzing only the movement of the container up to a certain moment), or at the end of the simulation. The user may also take a screenshot of the 3D view at any time, using the "Take Screenshots" button, and save it to a file.

Figure 7.7 shows the placement of the boxes at the end of a simulation. The boxes highlighted by the red circle are of particular interest, because they have moved and fallen in the process. This is easily visible when directly comparing Figure 7.6 and Figure 7.7. However, when using the tool, it is possible to see the movement of the boxes in real time, or by checking the "black dots" on the face of each box.

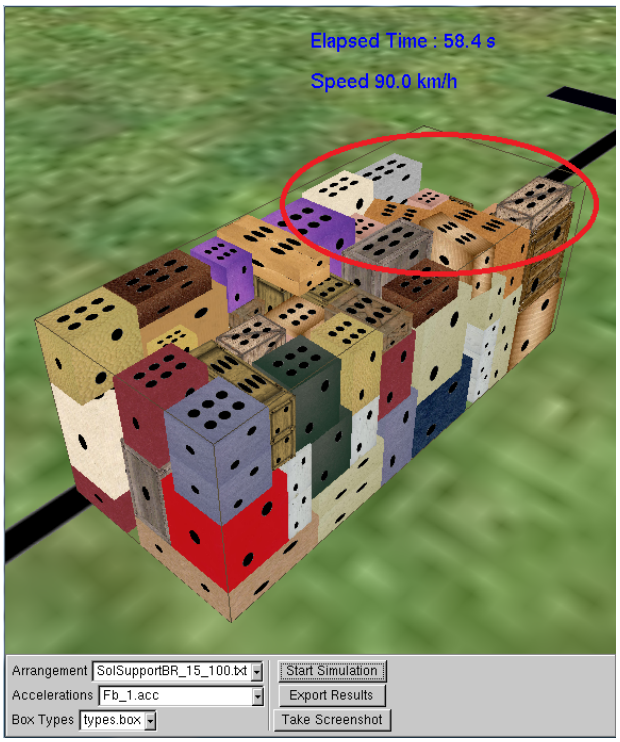


Figure 7.7: Final cargo arrangement after stabilization

7.3.2 Input files

The input files consist of the *Configuration*, *Cargo Arrangement*, *Acceleration* and *Box Types* files.

The *Configuration* file details several simulation configuration settings that must be set prior to running the tool itself. The parameters consist of:

- [*Physics Engine*]
  - *DensityValue* – represents the density of the boxes in kg/m^3.
  - *ConstantDensity* – defines whether the *DensityValue* is used to determine the mass of each box.
  - *SimulationStepsPerSecond* – sets the time period for updating the simulation in steps/s.
- [*Renderer Engine*]
  - *DrawAxis* – used to define if the Axes are to be drawn by the renderer.
  - *TexturePack* – address of the folder that contains the texture pack images, to be used for skinning the boxes with their respective material.

  - *UseRenderer* – used to enable or disable the graphical visualization. It can be disabled to perform batch simulations, without the need for visual feedback or interaction.
  - *BoxesOverlay* – used to enable or disable the use of shaders that render dice marks on the faces of the boxes, which help the user visualize if a box rotated from its original position.

- [*Export Settings*]
  - *ResultsIntervalInSeconds* – period of time, in s, for the simulator to sample the statistics of each box.
  - *SavePath* – address of the folder where the results will be saved.
  - *DetailedExportation* – used to enable or disable the exportation of detailed information.

- [*Simulation Settings*]
  - *StoppingCondition* – used to define the condition to end the simulation, after all the forces have been applied, either by "timeout" (x s) or by "sleeping" (no box has moved or rotated significantly).
  - *TimeoutInSeconds* – the timeout value in s, to be used if it is the chosen stopping condition.
  - *SleepingThreshold* – the movement threshold to be used in order to consider that the simulation has ended.
  - *BatchSimulation* – used to specify if the simulation is a batch simulation, that is, if there will be an attempt to pair all *Accelerations* files and *Cargo Arrangement* files in a folder. This allows for multiple simulations to be done without human intervention.
  - *BatchPath* – address of the folder containing all *Accelerations* and *Cargo Arrangement* files.
  - *DropThreshold* – used to set the minimum value, in m, a box has to shift from its initial position along the vertical axis, so that it can be considered to have fallen.
  - *DropLowerThreshold* – used to set the maximum value a box has to shift, in m, from its initial position, in order to be considered as having fallen.
  - *ContainerFriction* – friction of the container.
  - *GroundFriction* – friction of the ground.
  - *BoxFriction* – friction of the boxes.

- *LinearAccelerationThreshold* – used to set the maximum acceleration value, in m/s^2, a box can have before being flagged as exceeding this parameter.
- *WaitingTime* – the time period, in s, of the simulation pause between loading the container and applying the first forces (if available).

The *Cargo Arrangement* file (Figure 7.8) represents a possible loading scenario of boxes inside a container. It consists of a text file with a one-line header that represents the container volume and the total volume of the boxes, and a multi-line body, where each line represents the position of a box (through the 3D coordinates, in cm, of two diagonally opposing corners) and the type of box (identifier of material).

```
26281964    18642852
0   0   0   80   59   106   1
80  0   0   160  59   106   1
80  0   106 160  59   212   1
```

Figure 7.8: Example of a *Cargo Arrangement* file

The *Accelerations* file describes the accelerations the container is subject to during a period of time. Each line specifies an initial time in s, duration in s, and 3D acceleration vector in m/s^2 (Figure 7.9).

```
#Initial-Time / Duration / X / Y / Z
0      1     0   0   0
1      0.3   1   0   0
1.3    12    2   0   0
13.3   0.35  2   0   0
```

Figure 7.9: Example of an *Accelerations* file

The *Box Types* file describes the different physical properties that each type of box has. These properties override the ones specified, globally, in the *Configuration* file. The *Box Types* file has a similar structure to the *Accelerations* file, and, as such, it specifies the following attributes for each type of box per line of file:

- *Type-id* – id of the type of boxes to which the following properties will be applied.
- *Density* – density of the material the box is made of, in kg/m^3.
- *Friction* – the friction of the box material.
- *LinearAccelerationThreshold* – used to set the maximum acceleration value, in m/s^2, the type of box can have before being flagged as exceeding this parameter.

7.3.3 Output files

There are three output files: *Simulation Result*, *Simulation Summary* and *Batch Summary* file.

Simulation Result files, are automatically named with the template [Simulation]$Ar rangementFile$Acceler ationsFile$BoxTypesFile. The *Simulation Result* file consists of raw data for each box with multiple readings per box (as specified by the *Configuration* file), extracted from the Bullet physics engine. It contains the following data:

- *Id* – identifier of the box. It matches the line of the *Cargo Arrangement* file in which the box was declared.
- *Centre of mass position (X,Y,Z) displacement* – displays the container-relative box displacement, that is, the difference between the current 3D position and the initial position of the centre of mass, in relation to the position of the centre of mass of the container, in m.
- *Angular Velocity (X,Y,Z)* – 3D vector that contains the angular velocity of the box in rad/s.
- *Velocity (m/s)* – the current velocity of the box, in m/s.
- *Linear Acceleration* (m/s^2) – shows the current linear acceleration the box has in relation to the movement of the container, in m/s^2.
- *Elapsed time (s)* – represents each sampling interval time, in s.

Simulation Summary files are automatically named with the template [SUMMARY]$Arr angement File$AccelerationsFile$BoxTypesFile and contain information extracted from the *Simulation Result* file data:

- *Container Volume* – volume of the container, in cm^3.
- *Total Box Volume* – sum of the volume of all the boxes, in cm^3.
- *Number of Fallen Boxes* – by comparing the vertical displacement each box suffered during the simulation with the *Configuration* file DropThreshold value, it is possible to estimate the number of fallen boxes.
- *Volume of Fallen Boxes* – total volume of the boxes that have fallen.
- *Number of Boxes Exceeding Acceleration Limits* – number of boxes that exceed the LinearAccelerationThreshold value, specified in the *Configuration* or *Box Types* file.
- *Volume of Boxes Exceeding Acceleration Limits* – total volume of the boxes that exceed the LinearAccelerationThreshold value, specified in the *Configuration* or *Box Types* file.

- *Number of Boxes Exceeding the DBC* – number of boxes that were within the DBC damage area (boxes that have theoretically suffered damage).

- *Volume of Boxes Exceeding the DBC* – total volume of boxes that were within the DBC damage area (boxes that have theoretically suffered damage), in cm^3.

- *Centre of Mass Displacement* – difference between the initial and final position of the centre of mass, for each box, in cm.

- *Exceeded Acceleration Limits* – a flag, for each box, that specifies if the box exceeded the acceleration limit (1 or 0).

- *Exceeded DBC Limits* – a flag, for each box, that specifies if the box was within the DBC damage area (1 or 0).

The *Batch Summary* file, is named with the template [BATCH_ SUMMARY]$BatchDirectory, is only created if the application is set to run in batch mode. It summarizes the most important values for each simulation ran in batch mode, already present in the respective *Simulation Summary* files. It contains:

- *Cargo Arrangement file name*

- *Accelerations file name*

- *Box Types file name*

- *Container volume*

- *Total volume of boxes*

- *Number of fallen boxes*

- *Total volume of fallen Boxes*

- *Number of boxes exceeding the linear acceleration threshold*

- *Total volume of boxes exceeding the linear acceleration threshold*

- *Number of boxes exceeding the DBC*

- *Total volume of boxes exceeding the DBC*

7.4 Benchmark tests

The purpose of the physics simulation tool described here is to simulate the movement of a set of boxes inside a shipping container, when subject to external forces in typical extreme cases, such as vehicle full braking, tight cornering or fast lane changing.

Because friction is believed to be a relevant parameter to evaluate the physics engine performance, a set of benchmark tests were performed with friction as the main parameter. Numerical results obtained using the physics simulation tool and state-of-the-art engineering simulation software (Abaqus Unified FEA) were then compared with the analytical results, making it possible to assess the software packages' ability to model the friction phenomenon.

Abaqus FEA is a software suite for finite element analysis and computer-aided engineering. This software suite consists of five core software products:

- Abaqus/CAE, a software application used for pre-processing and visualizing the finite element analysis result.
- Abaqus/Standard, a general-purpose Finite-Element analyzer that employs the implicit integration scheme.
- Abaqus/Explicit, a special-purpose Finite-Element analyzer that employs an explicit integration scheme to solve highly nonlinear systems with many complex contacts under transient loads.
- Abaqus/CFD, a Computational Fluid Dynamics software application.
- Abaqus/Electromagnetic, a software application which solves advanced computational electromagnetic problems.

This product suite is used in academic work, as well as in industrial research projects, namely in the aerospace and automotive industry. In automotive engineering, it can be used to analyze sophisticated nonlinear engineering problems, such as impact/crash events, multibody systems, full vehicle loads and dynamic vibration.

Rigid elements were selected for the boxes as well as for the container floor in order to perform the benchmark tests with Abaqus FEA. Abaqus/Explicit was selected to perform the analysis, with a fixed time increment value of 0.1 ms. The possibility of contact between all surfaces was considered. Displacement, velocity and acceleration values for the centre of gravity of the box were recorded during all the simulations, which made post-processing the results quite simple.

7.4.1 Friction equations

Friction can be defined as the phenomenon of resistance of a body on another body, which delays or prevents relative movement between them. The force that expresses this resistance always acts tangent to the contact surface. Being a force between two bodies,

naturally conforms to the principle of action/reaction. The direction of the reaction force on a body that tends to move in a given direction is always opposite to that direction. In general, there are two types of friction: fluid friction, where surfaces are interleaved with a fluid layer (for instance, an oil), and dry friction, where two bodies are in direct contact.

In dry friction, if a force F acts on a block of weight W that is at rest, it generates reaction forces distributed along the contact surface between the two bodies. These forces have tangential or friction components T, and normal components N (see Figure 7.10). In general, there are two types of dry friction, static and kinetic friction. When two solid bodies that are not moving relative to each other, friction is static. When the two objects are moving relative to each other, friction is kinetic. Both the Bullet physics engine and Abaqus make no distinction between these two types of friction.

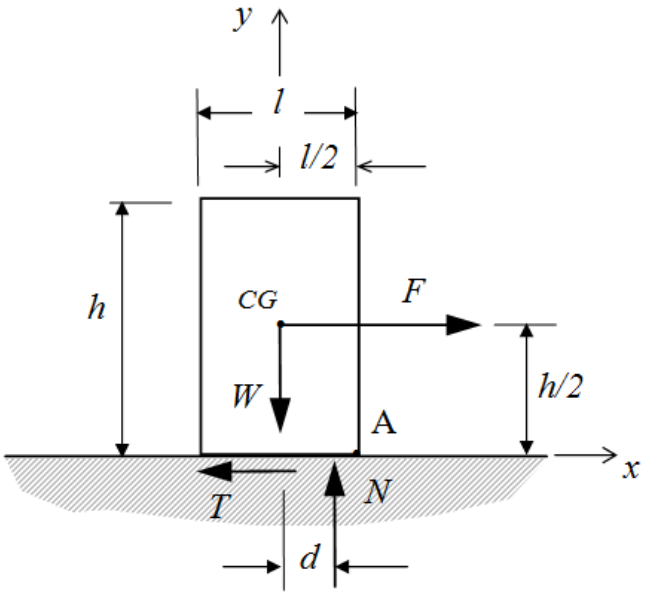


Figure 7.10: Forces acting on a box

Friction force T is independent of the surface area of contact, but depends directly on the resulting normal force N. The coefficient of static friction, μ_s, and the coefficient of kinetic friction, μ_k, between two surfaces in contact are determined experimentally. It is considered that the body is in the imminence of sliding if condition (7.1) is met, and in the imminence of tipping about A (see Figure 7.10) if condition (7.2) is met.

$$\begin{cases} T = \mu_s N \\ d < \frac{l}{2} \end{cases} \tag{7.1}$$

$$\begin{cases} T < \mu_s N \\ d = \frac{l}{2} \end{cases} \tag{7.2}$$

The body is considered to be translating with an acceleration a in a direction parallel to the horizontal plane if condition (7.3) is met.

$$\begin{cases} T = \mu_k N \\ F > T \\ d < \frac{l}{2} \end{cases} \tag{7.3}$$

Three tests were performed to test dry friction. The purpose of the first was to test condition (7.1), that is, the imminence of body sliding; the goal of the second was to test condition (7.2), that is, the imminence of body tipping; and the purpose of the third was to test condition (7.3), that is, the movement of the body when sliding occurs.

7.4.2 Sliding test

In order to test the sliding of a body, one box with dimensions 25 cm × 110 cm × 43 cm was placed on a horizontal plane. A coefficient of static friction was assigned between the body and the plane, and the force applied to the box parallel to the plane was incremented until the box started sliding. The acceleration in imminence of sliding on a horizontal plane is equal to (7.4).

$$a = \mu_s g \tag{7.4}$$

Figure 7.11 shows the values of the acceleration in the imminence of sliding for the range of coefficients of static friction between 0.1 to 0.8. The analytically calculated value is also represented. Both Abaqus and StableCargo provided results with high approximation to analytical values.

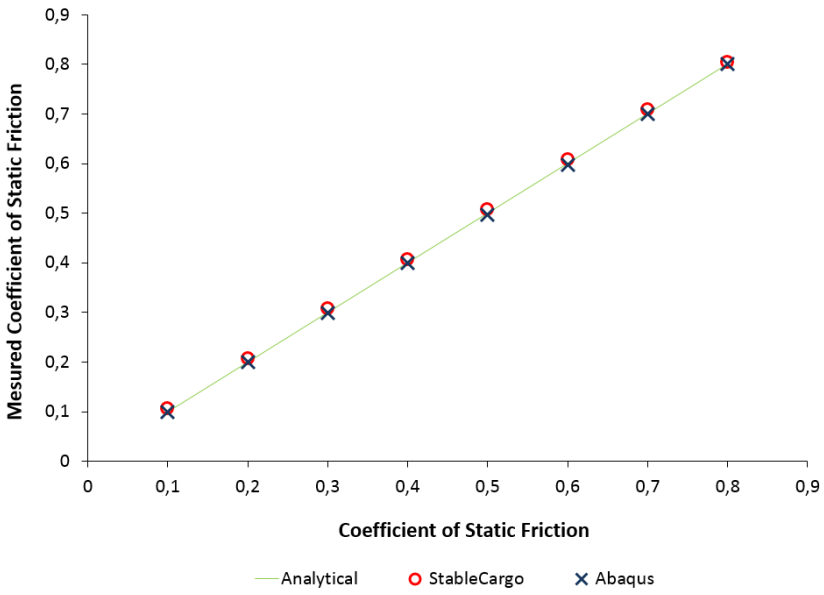


Figure 7.11: Measured coefficient of static friction

7.4.3 Tipping Test

Another test was conducted in order to analyze the imminence of body tipping. A coefficient of static friction between the body and the plane was assigned, and a force, parallel to the plane, was applied to the centre of gravity of the box. This force was incremented until the box started to move. If sliding occurred, then the coefficient of static friction was incremented and the test repeated. When tipping of the body occurred, the coefficient of static friction used was considered the measured coefficient of static friction. To guarantee that there is no sliding prior to tipping, the height and length ratio of the box must satisfy condition (7.5).

$$\frac{l}{h} > \mu_s \tag{7.5}$$

Figure 7.12 shows the values of the measured coefficient of static friction in the imminence of tipping for different values of the height and length ratio, ranging from 0.2 to 0.8.

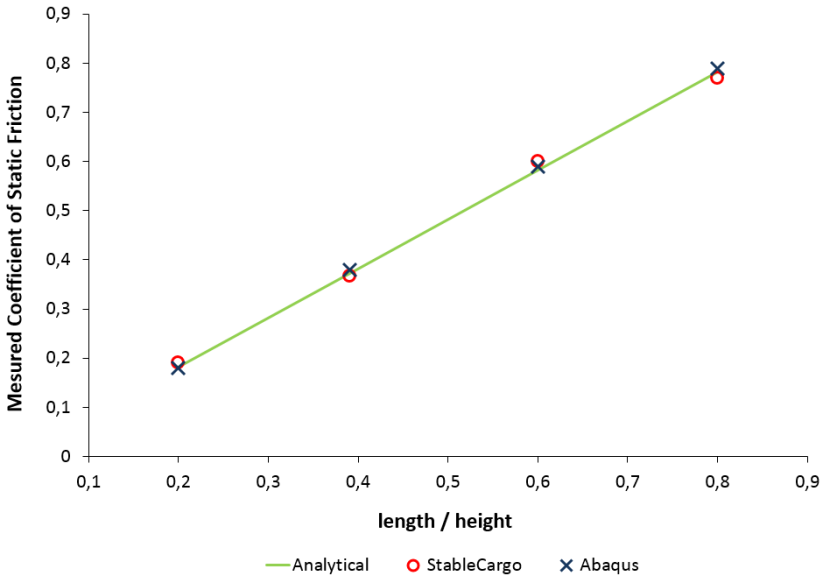


Figure 7.12: Measured coefficient of static friction for tipping

The analytical values are also represented. Results, obtained with Abaqus and StableCargo, are in line with the analytical values.

7.4.4 Movement test

To analyze the movement of a body, one box with dimensions 25 cm × 110 cm × 43 cm was placed on a horizontal plane. A coefficient of kinetic friction between the body and the

plane was assigned, and the force F was applied to the box, in a direction parallel to the horizontal plane, as shown in Figure 7.13.

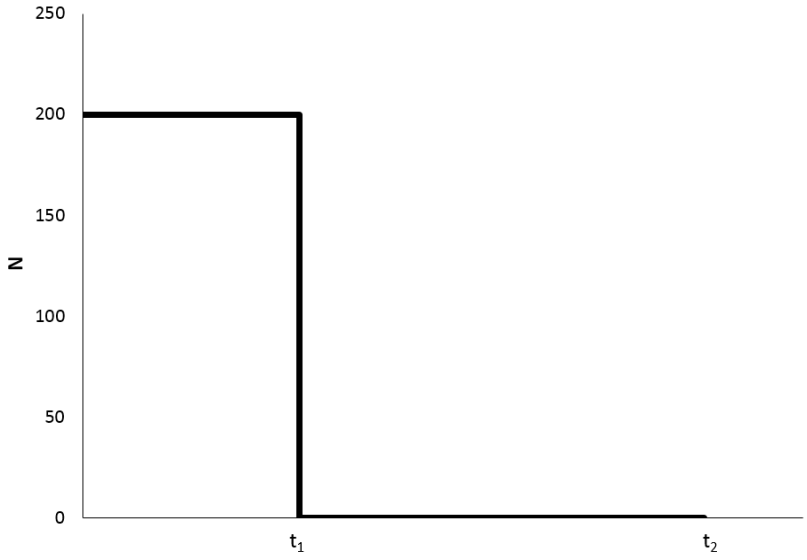


Figure 7.13: Force applied to the box

The mass of the box was set to 23.65 kg, the coefficient of kinetic friction between the box and the floor was set to 0.3, and the time values chosen were $t_1 = 0.6\,\text{s}$ and $t_2 = 1.8\,\text{s}$, which allowed the box to accelerate to 3.31 m/s in the first time interval and to decelerate to a complete stop at the end of the second time interval.

The acceleration in a horizontal plane in test (7.3) is equal to (7.6).

$$\begin{cases} a = \frac{F}{m} - \mu_k g & 0 < t < t_1 \\ a = -\mu_k g & t_1 < t < t_2 \end{cases} \tag{7.6}$$

Figures 7.14, 7.15 and 7.16 show the values obtained for the displacement, velocity and acceleration of the box. Results obtained with Abaqus and StableCargo are completely in line with the analytical values.

7.5 Conclusions and future work

Friction can be considered one of most relevant parameters when analyzing a physics engine. To evaluate the physics simulation tool described here in order to understand its ability to model friction, a set of benchmark tests were performed, and the results obtained were compared to analytical values and to those obtained using the state-of-the-art engineering simulation software Abaqus FEA.

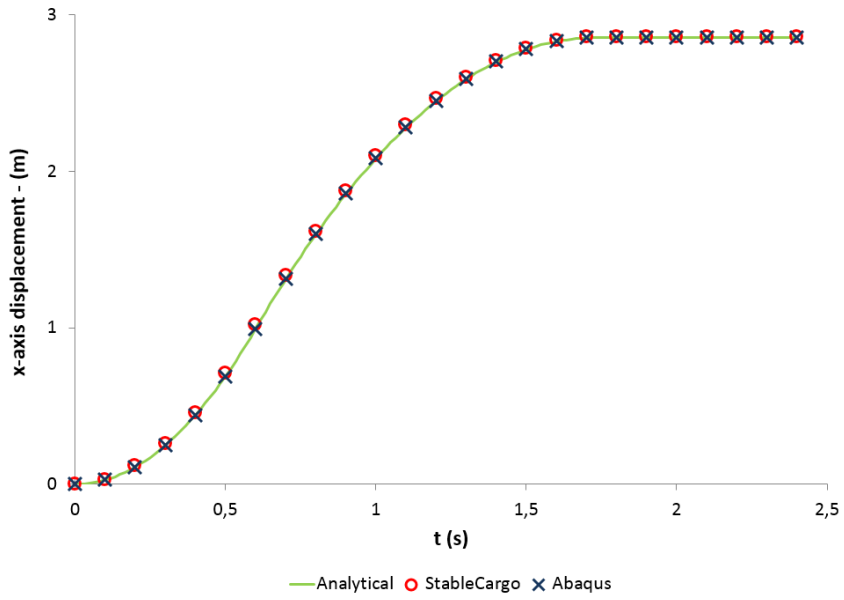


Figure 7.14: Displacement values for the box

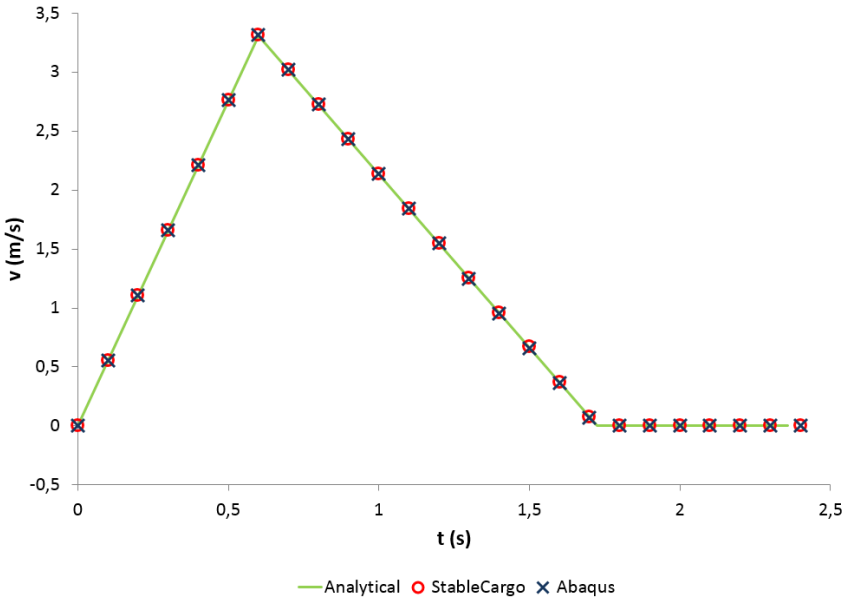


Figure 7.15: Velocity values for the box

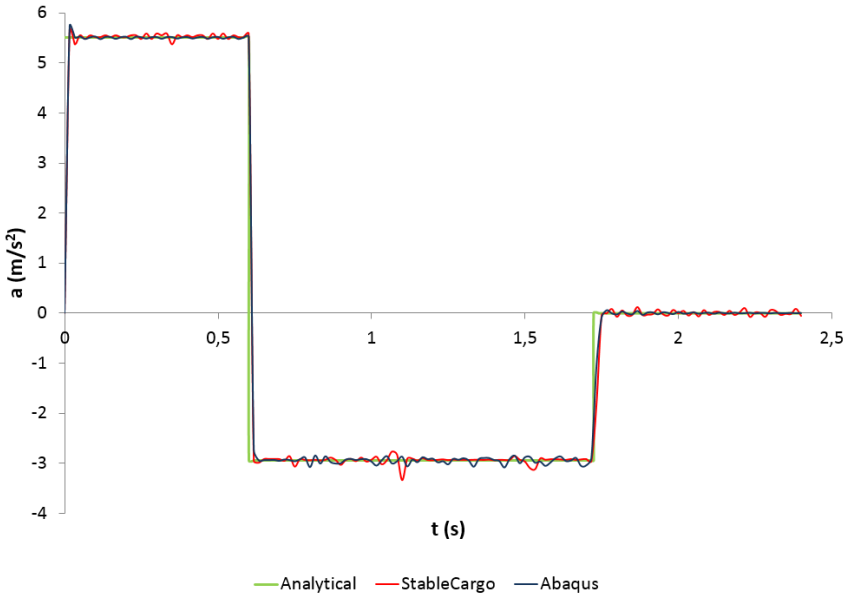


Figure 7.16: Acceleration values for the box

The results obtained are in agreement with both the analytical values and those obtained using Abaqus FEA, offering good prospects for the use of the tool to evaluate dynamic stability within the CLP.

Future work should concentrate on evaluating the physics simulation tool to model other events, such as collision and rebound, which may occur in a shipping container subject to typical extreme situations, such as vehicle full braking, tight cornering and fast lane changing. A set of benchmark tests involving these phenomena should be performed and the results compared to those obtained experimentally, and to other simulation engineering software, such as Abaqus FEA.

Chapter 8

Dynamic stability metrics for the container loading problem

A. Galrão Ramos[†] [‡], José F. Oliveira[†], José F. Gonçalves[§] and Manuel P. Lopes[‡]

[†] INESC-TEC and Faculty of Engineering, University of Porto
[‡] CIDEM, School of Engineering, Polytechnic of Porto
[§] LIAAD, INESC-TEC and Faculty of Economics, University of Porto

Abstract

The Container Loading Problem (CLP) literature has traditionally evaluated the dynamic stability of cargo by applying two metrics to box arrangements: the mean number of boxes supporting the items excluding those placed directly on the floor (M1) and the percentage of boxes with insufficient lateral support (M2). However, these metrics, that aim to be proxies for cargo stability during transportation, fail to translate real-world cargo conditions of stability. Even more relevant is the fact that dynamic stability itself has never been precisely defined, measured or quantified. In other words, when looking at cargo arrangements after transportation there is no evaluation criterion for the consequences of instability. In this paper two dynamic stability performance indicators are proposed to evaluate cargo arrangements after transportation: the number of fallen boxes (NFB) and the number of boxes within the Damage Boundary Curve fragility test (NB_DBC). Using 1500 solutions for well-known problem instances found in the literature, these new stability performance indicators are determined using a physics simulation tool (StableCargo), which replaces the real-world transportation by a truck with a simulation of the dynamic behaviour of container loading arrangements. The calculation of the NFB and the NB_DBC performance indicators requires the simulation's evolution over time. The

computational expense involved in this calculation leads us to propose two new proxy metrics for dynamic stability. The metrics are models of the proposed stability performance indicators, computed by multiple linear regression, which can be integrated within any container loading algorithm. Pearson's r correlation coefficient was used as an evaluation parameter for the performance of the models. The extensive computational results show that the proposed metrics are better proxies for dynamic stability in the CLP.

8.1 Introduction

The Container Loading Problem (CLP) is a real-world driven, combinatorial optimization problem which belongs to the more generic combinatorial optimization class of Cutting and Packing problems. The CLP addresses the optimization of the spatial arrangement of cargo inside containers or transportation vehicles, maximizing the usage of space. In the CLP, a set of boxes (small items) of parallelepiped shape must be packed orthogonally in a set of containers (large objects) of parallelepiped shape, in such a way that the boxes do not overlap and all the boxes of the subset lie entirely within the container. Each box can be associated with a box type $k(k = 1, ..., K)$, which is characterized by its depth, width and height (d_k, w_k, h_k) and required quantity n_k. Each container can be associated with a container type C_j, characterized by its depth, width and height (D_j, W_j, H_j) and available quantity m_j.

The typology proposed by Wäscher et al. (2007) for Cutting and Packing problems, classifies problems according to dimensionality, assortment of large items, assortment of small items, kind of assignment and shape of small items. The assortment of large and small items is related to the number of different types of items in the problem and is commonly classified either as strongly heterogeneous (many types) or weakly heterogeneous (few types). As an assignment problem, it can have two basic objectives: the maximization of the value of the cargo loaded, when the number of containers is not sufficient to accommodate all the cargo, or the minimization of the cost of containers, when there are sufficient containers to accommodate all the cargo.

The above defined CLP can be seen, in its essence, as a geometric assignment problem. Nonetheless, as it is a problem driven by real-world considerations, the solutions will be of limited applicability to real-world scenarios if real-world constraints are not considered. Using the work of Bischoff et al. (1995) as a starting point, Bortfeldt and Wäscher (2013) recently reviewed the literature on container loading constraints and suggested a classification for the real-world constraints. They distinguish between: container-related constraints (weight limits, weight distribution); item-related constraints (loading priorities, orientation, stacking); cargo-related constraints (complete shipments, allocation); positioning constraints; and load-related constraints (stability, complexity).

The stability constraint was referred to as one of the most important, being addressed in 37.4% of the articles reviewed by Bortfeldt and Wäscher (2013)(surpassed only by the orientation constraint with 70.6%). Its high relevance reflects the impact on customer satisfaction and operational efficiency as well as the safety of both workers involved in

loading operations and other persons or vehicles during transportation. These authors also note that there are two types of stability constraints, distinguishing between static and dynamic stability, and that only a small number of papers address dynamic stability.

This paper focus on cargo dynamic stability. Its aim is to propose a new set of dynamic stability metrics for the CLP, reflecting real-world dynamic stability, to be used within a container loading algorithm.

The proposed approach to determine the new stability metrics is illustrated in Figure 8.1. Firstly, two dynamic stability performance indicators are derived and proposed. These indicators try to measure and quantify the effects of instability in cargo arrangements after transportation takes place. Secondly, instead of the actual transportation process by trucks, a set of transportation scenarios is implemented in a specialised physics simulation tool (StableCargo). The simulation uses 1500 cargo arrangements, corresponding to solutions previously obtained for the well-known Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999) CLP instances, employing the selected transportation scenarios; the outcome is then evaluated in terms of instability consequences by looking at the two dynamic stability performance indicators. Thirdly, measuring dynamic stability by transporting the cargo or running a simulation model is time consuming and infeasible to integrate into CLP algorithms that require the (partial or full) evaluation of thousands of solutions. Thus, a set of characteristics is proposed for the cargo and its arrangement which may influence its behaviour in terms of stability during transportation. Finally, two new metrics for dynamic stability are proposed which are applicable to cargo arrangements and capable of being integrated into CLP algorithms. These metrics are derived from a multiple linear regression analysis that models the relation between the two dynamic stability performance indicators and the set of characteristics for the cargo and its arrangements.

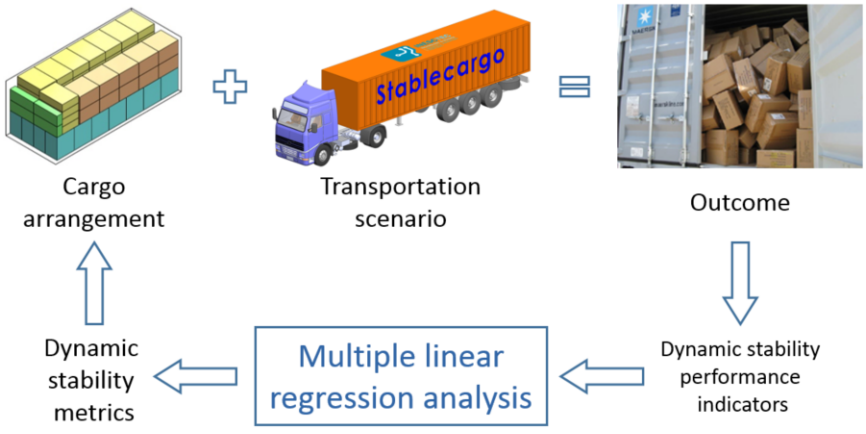


Figure 8.1: Approach to determine dynamic stability metrics

The remainder of the paper is organized as follows. Section 8.2 presents an overview of the literature covering dynamic stability within the CLP and identifies the existing dynamic stability metrics. Section 8.3 starts by presenting the new cargo dynamic stability performance indicators, followed by the StableCargo physics simulator. In Section 8.4,

the multiple linear regression models are presented, as well as the new set of geometric metrics used as input variables to the models. Section 8.5 reports the results from the computational experiments. Finally, Section 8.6 draws some conclusions from the findings.

8.2 Literature review

According to Bortfeldt and Wäscher (2013), there are two different types of stability in the CLP: static and dynamic stability. Static stability refers to the stability of cargo during loading operations and dynamic stability refers to the stability of cargo during transportation. While there are authors that clearly distinguish the two types, others do not make such distinction and treat stability as a single constraint. The majority of approaches found in literature only focus on static stability (Bortfeldt and Wäscher, 2013).

The approaches to dynamic stability found in the CLP literature can be classified according to the two types of container wall: flexible and rigid wall.

- **Flexible container wall** - A flexible container wall does not provide lateral support for the boxes. This is the case for the three-dimensional pallet loading problem where the walls of the container are virtual and the goal of the pallet arrangement is to obtain a pallet that after being shrink wrapped can be treated as a single rigid body. These problems focus on ensuring that the boxes interlock as well as possible, therefore avoiding the possibility of generating guillotine cuts, since it is considered that they contribute to an unstable arrangement. Carpenter and Dowsland (1985) proposed two stability criteria to measure dynamic stability. One criterion states that each box must have its base in contact with at least two boxes, ignoring contact surfaces areas with less than a predetermined percentage value of the base of the box, thus providing a measure of the degree of interlocking of the pallet. Another criteria considers the problems related with vertical pallet guillotine section cutting. The criteria states that the guillotine cut must not exceed a predetermined percentage value of the maximum length or width of the stack. Bischoff (1991) replaced the first criterion by another one where each box positioned on the perimeter of the pattern has to be supported by at least two boxes in the layer below, ignoring contact surface areas with less than a predetermined percentage value of the base of the box. While Carpenter and Dowsland (1985) and Bischoff (1991) use these criteria to evaluate the dynamic stability of generated cargo arrangements, other authors (e.g. Abdou and Elmasry (1999)) developed approaches that enforce interlocking for the column stacks, but do not measure it. It is worthwhile mentioning that there are also containers with flexible walls (curtains) where the reasoning presented above also applies when solving the actual CLP.

- **Rigid container wall** - In problems with rigid container walls, the walls of the container provide lateral support for the boxes and contribute to the dynamic stability of the cargo. Bischoff et al. (1995) presented three metrics for evaluating dynamic stability of cargo. The first metric, measuring the interlocking between items of a

cargo arrangement, is the average number of supporting boxes for each box that is not positioned on the container floor (M1). The higher the value of M1 the better. The second metric is similar to the first but does not consider contact areas with less than 5% of the base area of a box. The third metric, which focus on the lateral movement of boxes, records the percentage of boxes which do not have at least three of their four lateral sides in contact with another box or with the container walls (M2). The lower the value of M2 the better. The first metric (M1) and third metric (M2) are the ones that are mainly used within the CLP to evaluate dynamic stability.

The problem tests instances most frequently used to evaluate the effectiveness of an algorithm for the single CLP, are the 1500 problem instances proposed by Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999). These instances are organized in 15 classes, with a total of 100 instances per class. They are designated here as BR1 - BR15.

As far as the authors are aware, only five authors have evaluated their container loading solutions with the M1 metric and nine with the M2 metric. The results are presented in Table 8.1 and Table 8.2, respectively. The column headings for the computational results in the tables below refer to the different algorithms used, namely: H_B_al—an heuristic approach of Bischoff and Ratcliff (1995b); H_BR—an heuristic approach of Bischoff et al. (1995); GA_GB—a genetic algorithm of Gehring and Bortfeldt (1997); H_E—an heuristic approach of Eley (2002); H_B—an heuristic approach of ; GRASP_MO— a GRASP algorithm of Moura and Oliveira (2005); AAR1 and ARR2 — two constructive heuristics of de Araújo and Armentano (2007); TS_L_al—an hybrid tabu search approach of Liu et al. (2011a). The H_E and TS_L_al results were taken from a graphical representation, therefore they might not be entirely accurate.

Table 8.1: Algorithm performance for the M1 metric

M1 Problem	H_B_al (1995)	H_BR (1995)	H_B (2003)	GRASP_MO (2005)	AAR1 (2007)	AAR2 (2007)
BR1	**2.02**	1.13	1.17	1.07	1.15	1.18
BR2	**2.22**	1.10	1.14	1.10	1.15	1.18
BR3	**2.20**	1.08	1.09	1.09	1.12	1.15
BR4	**2.10**	1.07	1.07	1.10	1.11	1.13
BR5	**2.09**	1.06	1.06	1.10	1.11	1.13
BR6	**2.04**	1.06	1.05	1.10	1.10	1.12
BR7	**1.92**	1.04	1.03	1.11	1.08	1.10
BR8	-	-	-	**1.12**	-	-
BR9	-	-	-	**1.10**	-	-
BR10	-	-	-	**1.10**	-	-
BR11	-	-	-	**1.14**	-	-
BR12	-	-	-	**1.15**	-	-
BR13	-	-	-	**1.16**	-	-
BR14	-	-	-	**1.16**	-	-
BR15	-	-	-	**1.17**	-	-
Mean (BR 1-7)	**2.08**	1.08	1.09	1.10	1.12	1.14
Mean (BR 8-15)	-	-	-	**1.14**	-	-
Mean (BR 1-15)	-	-	-	**1.12**	-	-

∗ **The best values appear in bold**

Table 8.2: Algorithm performance for the M2 metric

M2 Problem	H_B_al (1995)	H_BR (1995)	GA_GB (1997)	TS_BG (1998)	H_E (2002)	H_B (2003)	GRASP_MO (2005)	AAR1 (2007)	AAR2 (2007)	TS_L_al (2011)
BR1	8.5	10.4	11.0	13.0	9.8	12.4	11.5	**6.0**	10.8	7.4
BR2	11.2	14.6	16.0	19.0	13.5	15.3	12.7	**9.2**	13.7	11.6
BR3	15.9	19.7	18.5	24.5	18.0	17.1	17.8	**10.4**	16.2	15.2
BR4	17.5	23.5	21.5	29.9	20.5	18.7	20.0	**12.5**	18.1	17.8
BR5	21.6	26.0	22.5	34.0	21.5	20.8	22.8	**13.7**	19.5	19.2
BR6	22.1	31.0	25.0	33.5	22.9	23.3	26.5	**15.7**	20.9	22.2
BR7	27.1	36.0	28.5	46.1	26.0	24.6	28.8	**18.2**	23.9	25.2
BR8	-	-	-	-	-	-	**32.8**	-	-	-
BR9	-	-	-	-	-	-	**37.5**	-	-	-
BR10	-	-	-	-	-	-	**39.2**	-	-	-
BR11	-	-	-	-	-	-	**40.6**	-	-	-
BR12	-	-	-	-	-	-	**41.4**	-	-	-
BR13	-	-	-	-	-	-	**41.7**	-	-	-
BR14	-	-	-	-	-	-	**43.1**	-	-	-
BR15	-	-	-	-	-	-	**44.1**	-	-	-
Mean (BR 1-7)	17.7	23.0	20.4	28.6	18.9	18.8	20.0	**12.2**	17.6	16.9
Mean (BR 8-15)	-	-	-	-	-	-	**40.1**	-	-	-
Mean (BR 1-15)	-	-	-	-	-	-	**30.7**	-	-	-

∗ **The best values appear in bold**

For the M1 metric, it can observed that H_B_al provides the best results for problems BR1 to BR7, while for the same set of problems under the M2 metric the best results are provided by AAR1. For problems BR8 to BR15 only GRASP_MO provides results for M1 and M2. It is important to note that the objective function of these algorithms aims to maximise the volume of the cargo loaded in the container, and the dynamic stability constraint is treated as a soft constraint.

The solutions with the best average volume occupied in the container, for the BR1 - BR7 class instances, are from the AAR2 algorithm of de Araújo and Armentano (2007). Therefore, the algorithm with the best performance in terms of volume utilization is not the algorithm that has the best average values for either M1 or M2. This observation is significant since the literature mentions that there is a correspondence between stability and a high volume occupation of the container (Bortfeldt and Wäscher, 2013). Nevertheless, the percentage of volume occupied in the container will also be considered as a dynamic stability metric in the study presented here and will be referred to as %Vol.

8.3 New dynamic stability performance indicators for the CLP

One of the requirements of cargo transportation is to physically move products from a source to a destination, efficiently, without damaging the products and in a safe manner. It is of the utmost importance that when cargo is placed inside the transportation vehicle, this is done in such a way that it can endanger neither persons nor goods and cannot slide or fall off the vehicle. Safety during cargo transportation is a major concern, particularly

in road transport.

Two new dynamic stability performance indicators are proposed to evaluate dynamic stability within the CLP: the number of fallen boxes (NFB) and the number of boxes within the Damage Boundary Curve damage area (NB_DBC). The NFB is the number of boxes whose difference along the $z - axis$ between the initial and final position of its centre of gravity is greater than zero. These performance indicators translate what can be considered as one of the most costly consequences of a vertical position shift during transportation, that is, fallen cargo. The cost of the position shift may be seen in damage to the cargo as well a reduction in operations efficiency with more time needed to unload or for additional quality inspection. However, fallen cargo is not the only type of displacement that can cause damage. Horizontal impact between boxes, and between boxes and the container walls, can also result in damage to the cargo. Therefore, another metric is also proposed: the number of boxes within the Damage Boundary Curve (DBC) damage area. The DBC concept was introduced in the field of packaging design by Newton (1968) to evaluate product fragility.

The DBC concept assumes that a product will fail (suffer damage) when a combination of two factors occurs: there is both a change in velocity and an acceleration that are higher than the respective critical values the product can support. These two factors are plotted in a plane, with the velocity change as the horizontal axis and the acceleration peak as the vertical axis. The DBC divides the plane in two regions (see Figure 8.2), the Damage Region, where the combination of the values of velocity change and peak acceleration are enough to cause damage, and the No Damage Region for all other combinations (Newton, 1968). Even though the DBC can be determined from the product material and shape, it is usually determined experimentally in laboratory tests. A standard test is available in the ASTM D-3332 "Standard Test Methods for Mechanical Shock and Fragility of Products Using Shock Machines" (American Society for Testing Materials, 2010).

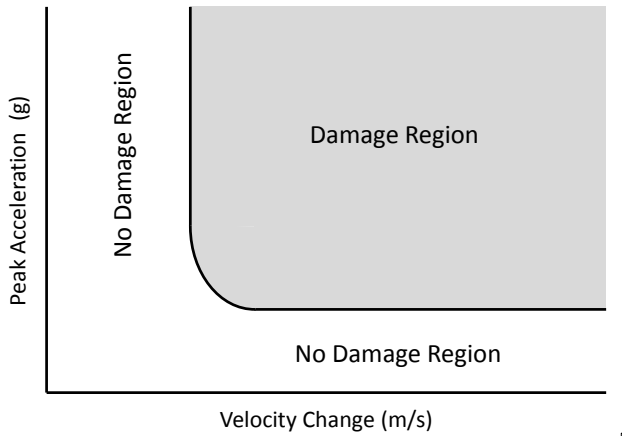


Figure 8.2: Typical damage boundary curve

8.3.1 StableCargo physics simulation tool

In this paper we consider that containers have a rigid wall and we focus on evaluating whether or not a cargo arrangement can be expected to cause damage to the products during transportation. Damage to the products during transportation can result from several dynamic forces that are caused by vehicle impacts and vehicle vibrations, resulting in the products experiencing mechanical shocks and mechanical vibrations during transportation.

A mechanical shock is the result of the impact between objects, such as the container and a box or two boxes, and occurs whenever there is a sudden variation in the velocity or acceleration of the objects (Brandenburg and Lee, 1985). In road transportation, they are the result of typical borderline vehicle manoeuvres, such as full braking, tight cornering, rapid lane changing and obstacle avoidance manoeuvres (Kaps, 2011).

A mechanical vibration is the repeated oscillation of a rigid body or mechanical system, in alternately opposite directions, about its static equilibrium position (Hibbeler, 2010a). For example, in road vehicle transportation, the several sources of mechanical vibration are a consequence of the vehicle motion and are classified in two categories: road roughness and on-board sources. The first category is related to the road profile and is the result of the deviation in the elevation of the vehicle while moving. The latter category encompasses the rotating components of the vehicle, such as the tyre/wheels, the driveline or engine (Gillespie, 1992).

Measurements of mechanical shock and mechanical vibration forces are usually determined through acceleration. The acceleration of an object is the rate at which its velocity changes over time and, according to Newton's second law of motion, the acceleration of an object depends directly upon the net force acting upon the object, and inversely upon the mass of the object.

The most common and meaningful way of expressing dynamic forces is as a multiple or submultiple of the force of gravity (Gillespie, 1992; Fiedler, 2009). Accordingly, it is possible to determine the resultant force acting upon an object indirectly through its acceleration. The acceleration of gravity is represented by the symbol "g" and its numerical value is defined as $9.81\,\mathrm{m/s^2}$.

To evaluate the possibility of damage to a product during transportation it is necessary to know the acceleration experienced by the container over time. Data for evaluating impacts and vibrations are typically collected using equipment that measures acceleration in the transportation environment (Young, 2009).

A shock to a packaged product usually lasts between 2 and 50 ms and can have a magnitude of up to 1000 g's in acceleration (Fiedler, 2009). These events reinforce the need to study the movement of the boxes located inside the container under the action of the externally induced acceleration without performing physical experimentation.

In this subsection, the simulation tool developed to analyse the dynamic stability of the cargo within the CLP is presented. Named StableCargo, it simulates the physical behaviour of the cargo in a container when different acceleration forces are applied to the container, much like those experienced in real life situations. This tool is based on the CGFLib library (a library for computer graphics based on OpenGL) for creating a

graphics user interface (GUI), and the Bullet physics engine (Coumans, 2013) that models the boxes and the container and the events that influence their behaviour.

A physics engine is a computer software designed to simulate various physical phenomena such as rigid body dynamics, soft body dynamics or fluid dynamics. It manages the forces applied to objects and the interactions between them by simulating Newtonian physics (Jones, 2011; Seugling and Rolin, 2006). The use of simulation tools based on physics engines to analyse the stability of cargo during transportation within the CLP was first presented in the work of Mustafee and Bischoff (2013). However these authors only performed some exploratory experiences with the physics engine Jinngine.

In the StableCargo simulation tool, the boxes and container are modelled as rigid bodies, that is, bodies that do not deform under the action of applied forces but move in response to external forces exerted on them. By analysing how the system of rigid bodies changes as a function of time it is possible to determine the value of the new performance indicators proposed to evaluate the dynamic stability of the cargo.

There are four input files to the StableCargo tool. The *Configuration* file includes a set of information configuration settings that must be set prior to running the tool itself. The *Accelerations* file describes the accelerations the container experiences during a period of time. The *Cargo Arrangement* file represents a cargo arrangement of boxes inside a container, that is, a CLP solution and the *Box Types* file describes the different physical properties of each type of box. Figure 8.3 depicts the StableCargo GUI after loading a cargo arrangement. After applying the chosen *Accelerations* file to a CLP solution, StableCargo generates information such as the number of fallen boxes or the displacement of the boxes during simulation. The StableCargo physics simulator tool was developed in C++ and a more detailed description of the StableCargo tool can be found in Ramos et al. (2014a).

8.4 Multiple regression analysis

The incorporation of the StableCargo physics simulation tool within a container loading algorithm, to compute the new dynamic stability performance indicators, is of limited applicability since it would make the algorithm computationally very expensive. This is directly related to the fact that it is mandatory that the physics simulation runs for a given amount of time to generate results. In order to be able to measure dynamic stability within a container loading algorithm without spending this amount of time, a multiple linear regression model is proposed for each of the new dynamic stability performance indicators that proxies the results of the simulation model. The coefficients of the regression model are tuned to adjust the behaviour of the model to the stability performance indicators, as assessed by the physics simulator.

The aim of a multiple linear regression analysis is to find a linear relationship between the dynamic stability performance indicators and several possible independent (explanatory) variables. In addition to the main effects of the independent variables, effects of the interaction between them were also included in the analysis. Its general form is given by:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_K x_K + \beta_{K+1} x_1 x_2 + ... + \beta_M x_{K-1} x_K + \varepsilon \tag{8.1}$$

Figure 8.3: Representation of a loaded *Cargo Arrangement* file

where y is the dependent variable, $\beta_m\,(m = 0, 1, ...M)$ are the regression coefficients, $x_k\,(k = 1, ..., K)$ are the independent variables and ε is the residual error.

A number of factors were hypothesized to influence the NFB and the NB_DBC. As observed in Subsection 8.3.1, besides the Newtonian physics, the outcome of a simulation is dependent on the acceleration scenario the container is subject to, the cargo arrangement and the physical properties of the boxes and the direction of the acceleration. As such, the independent variables selected for inclusion in the models are related with these factors.

The first properties of the boxes used in the regression model were the friction coefficient (FrC) and the density (DeN) (in kg/m^3). The acceleration (AcL) is determined by the maximum value (in g′s) of each acceleration scenario. The direction of movement (AxS) was characterized by the container dimension (in m) along the axis in which the movement occurs.

8.4.1 Independent variables used to represent cargo arrangement features

The cargo arrangement inside a container is probably one of the main elements in determining the outcome of a dynamic simulation. In the proposed models four types of variables are used to describe a cargo arrangement: the heterogeneity of the cargo, the percentage of container volume used, and two sets of four geometric variables, that is, the number of boxes with possible vertical displacement ($Num\Delta z$), and the sum of possible horizontal displacements ($HDisp$), along the x and y axes (in m).

The heterogeneity of the cargo (HeG) is measured by the number of different types of boxes inside the container, which influences the generation of a large compact block whenever the heterogeneity is weak. The percentage of container volume used (VoL) reflects the overall availability of space inside the container that can allow the movement of boxes.

The main principle behind the use of the proposed geometric variables is the measurement of the horizontal and vertical allowable movement of the boxes along the main direction of the acceleration (F), the opposed direction (B), the transverse left (L) and the transverse right (R).

To measure the allowable movement of a box we use a compaction process which tries to move each box as close as possible to the four edges (X and Y, backwards and forwards) of the container floor (one at a time). The compaction procedure tries to represent the results of full braking, tight cornering, or standing start full acceleration by first moving all of the boxes horizontally and then moving them in the downward vertical direction. The allowable movement of a box in each direction is measured by the difference between the final and initial position of the box. Figure 8.4 illustrates the four compacting directions.

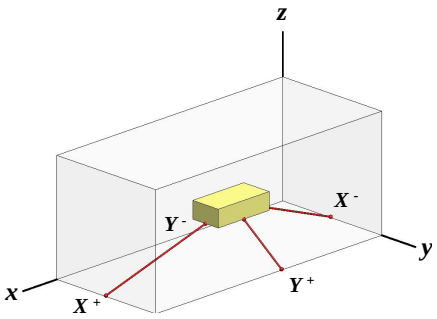


Figure 8.4: Illustration of the four compacting directions

The main procedure in the compaction of the cargo arrangement algorithm consists of the following two steps: a horizontal movement in the compacting direction and a downward vertical movement. For each of the four edges, the algorithm begins by ordering the boxes by non-decreasing order of the horizontal distance to the compacting edge.

Let the container be characterized by its depth, width and height, (D, W, H), and the position of a box b inside the container be given by its minimum and maximum coordinates, (x_1, y_1, z_1) and (x_2, y_2, z_2), respectively. We define the distance between a box and an edge $d(b, e)$ as the vector of two components (8.2), the horizontal and the vertical distance.

$$\begin{cases} (\ D - x_2\ ,\ z_1) & if \quad X^+ \\ (\ x_1\ ,\ z_1) & if \quad X^- \\ (\ W - y_2\ ,\ z_1) & if \quad Y^+ \\ (\ y_1\ ,\ z_1) & if \quad Y^- \end{cases} \tag{8.2}$$

For instance, if we are compacting in the X^- direction, and we have two boxes, $b_1 = (4, 3, 2), (6, 5, 3)$ and $b_2 = (5, 2, 3), (8, 4, 5)$, their distances to the edge are $d(b_1, X^-) = (4, 2)$ and $d(b_2, X^-) = (5, 3)$. Out of these boxes the one closer to the edge is b_1.

Next, according to the previously described order, the algorithm selects a box, one at a time, and moves it horizontally towards the edge, while there is empty space available. The next box is then selected and the procedure is repeated.

Considering the position of the boxes after the horizontal movement, the algorithm reorders the boxes by non-decreasing order of the vertical distance to the container floor. It then tries to move each box in the downward direction starting with the ones closer to the container floor. This alternation between horizontal and vertical movement is repeated until there are no boxes that can be moved any further.

Figure 8.5 illustrates the compacting procedure towards X^-. The algorithm for compacting the arrangement is described in Algorithm 6.

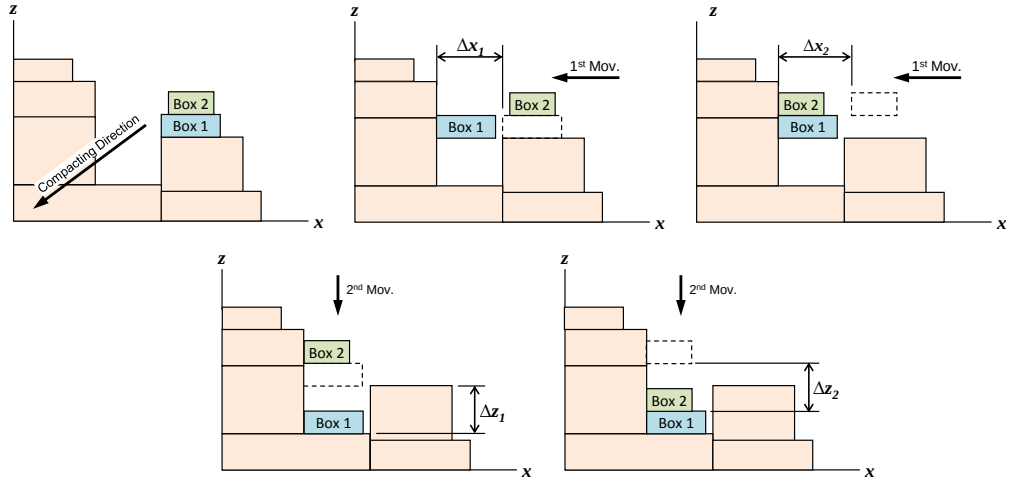


Figure 8.5: Example of the compacting procedure

Let (x_i, y_i, z_i) and (x_f, y_f, z_f) be the minimum coordinate of a box before and after compacting the cargo layout towards one of the edges of the container floor, respectively. After compacting the cargo arrangement, the allowable movement of each box is determined by (8.3).

$$\begin{cases} \Delta x_b^+ = x_f - x_i & if \quad X^+ \\ \Delta x_b^- = x_i - x_f & if \quad X^- \\ \Delta y_b^+ = y_f - y_i & if \quad Y^+ \\ \Delta y_b^- = y_i - y_f & if \quad Y^- \\ \Delta z_b = z_i - z_f \end{cases} \tag{8.3}$$

The number of boxes with vertical displacement ($Num\Delta z$) is determined by the number of boxes with $z_i > z_f$. The $Num\Delta z$ is measured in the four compacting directions.

$$Num\Delta z = \sum_b 1 | \Delta z_b > 0 \tag{8.4}$$

The total of the displacements in one direction is given by the sum of the displacement of each box whose displacement exceeds a predetermined minimum (Δmin) value.

Algorithm 6 Compacting of the arrangement (Loading arrangement M, Horizontal axis)

Input: Let M be a set of n boxes b_i, $i = 1, ..., n$
Output: Let $Compact\_M$ be the compacted cargo arrangement

Begin
Let *moved* be a boolean value where TRUE represents a movement of a box in a direction and FALSE the opposite case.
Let *direction* represent the direction of movement along one axis.
Let *direction_switch* represent a counter to switch the direction in which the boxes move.
$Compact\_M \leftarrow M$
$stop \leftarrow 0$
$direction\_switch \leftarrow 0$
while $(stop < 2)$ **do**
  if $direction\_switch = 0$ **then**
    $direction \leftarrow horizontal\_axis$
  else
    $direction \leftarrow z - axis^{-}$
  end if
  $moved \leftarrow$ FALSE
  Order $Compact\_M$
  for each box $b_i \in Compact\_M$ **do**
    Call MoveBox$(b_i, direction)$
    if b_i has moved **then**
      $moved \leftarrow$ TRUE
    end if
  end for
  if moved=FALSE **then**
    $stop++$
  else
    $stop \leftarrow 0$
  end if
  $direction\_switch++$
  if $direction\_switch = 2$ **then**
    $direction\_switch \leftarrow 0$
  end if
end while
End

The minimum displacement (Δmin) indicates the minimum distance a box must travel, subject to the maximum acceleration of the container, a_c, in order to reach an impact velocity v_c, that can cause damage to the box. The (Δmin) is determined by (8.5).

$$\Delta min = \frac{v_c^2}{2 \times a_c} \tag{8.5}$$

The $HDisp_{X+}$, $HDisp_{X-}$, $HDisp_{Y+}$, or $HDisp_{X-}$ values are given by the equations presented in (8.6)

$$\begin{aligned} HDisp_{X+} &= \sum_{b|\Delta x_b^+ > \Delta min} \Delta x_b^+ \\ HDisp_{X-} &= \sum_{b|\Delta x_b^- > \Delta min} \Delta x_b^- \\ HDisp_{Y+} &= \sum_{b|\Delta y_b^+ > \Delta min} \Delta y_b^+ \\ HDisp_{Y-} &= \sum_{b|\Delta y_b^- > \Delta min} \Delta y_b^- \end{aligned} \tag{8.6}$$

Table 8.3 and Table 8.4 present the relation between the compacting direction variables and the model variables according to the main direction of the acceleration scenario.

Table 8.3: Correspondence between model variables, acceleration direction and $Num\Delta z$ metrics

Model Variables	Acceleration direction			
	X^+	X^-	Y^+	Y^-
Fz	$Num\Delta z_{X+}$	$Num\Delta z_{X-}$	$Num\Delta z_{Y+}$	$Num\Delta z_{Y-}$
Bz	$Num\Delta z_{X-}$	$Num\Delta z_{X+}$	$Num\Delta z_{Y-}$	$Num\Delta z_{Y+}$
Lz	$Num\Delta z_{Y+}$	$Num\Delta z_{Y-}$	$Num\Delta z_{X-}$	$Num\Delta z_{X+}$
Rz	$Num\Delta z_{Y-}$	$Num\Delta z_{Y+}$	$Num\Delta z_{X+}$	$Num\Delta z_{X-}$

Table 8.4: Correspondence between model variables, acceleration direction and $HDisp$

Model Variables	Acceleration direction			
	X^+	X^-	Y^+	Y^-
Fd	$HDisp_{X+}$	$HDisp_{X-}$	$HDisp_{Y+}$	$HDisp_{Y-}$
Bd	$HDisp_{X-}$	$HDisp_{X+}$	$HDisp_{Y-}$	$HDisp_{Y+}$
Ld	$HDisp_{Y+}$	$HDisp_{Y-}$	$HDisp_{X-}$	$HDisp_{X+}$
Rd	$HDisp_{Y-}$	$HDisp_{Y+}$	$HDisp_{X+}$	$HDisp_{X-}$

Note that partitioning the analysis into the four directions allows an evaluation of the dynamic stability of the cargo according to the type of real experienced borderline vehicle impacts, such as full braking or tight cornering or, if required, a combination of several impacts.

8.4.2 Multiple regression models

Two general models are used to determine the dynamic stability metrics. The first model predicts the NFB, using FrC, DeN, HeG, AcL, VoL, AxS, Fz, Bz, Lz and Rz as independent variables along with the correspondent pair-wise interaction terms. An interaction occurs if the effect of an independent variable on the dependent variable depends on the level of another independent variable. The full NFB model is presented in (8.7). The second model predicts the NB_DBC, using FrC, DeN, HeG, AcL, VoL, AxS, Fd, Bd, Ld and Rd as independent variables along with the correspondent pairwise-interaction terms. The full NB_DBC model is presented in (8.8).

$$\begin{aligned} NFB = {} & \beta_0 + \beta_1 Fz + \beta_2 Bz + \beta_3 Lz + \beta_4 Rz + \beta_5 FrC + \beta_6 DeN + \\ & + \beta_7 HeG + \beta_8 AcL + \beta_9 VoL + \beta_{10} AxS + \beta_{11} Fz.Bz + \\ & + \beta_{12} Fz.Lz + \beta_{13} Fz.Rz + \beta_{14} Fz.FrC + \beta_{15} Fz.DeN + \\ & + \beta_{16} Fz.HeG + \beta_{17} Fz.AcL + \beta_{18} Fz.VoL + \beta_{19} Fz.AxS + \\ & + \beta_{20} Bz.Lz + \beta_{21} Bz.Rz + \beta_{22} Bz.FrC + \beta_{23} Bz.DeN + \\ & + \beta_{24} Bz.HeG + \beta_{25} Bz.AcL + \beta_{26} Bz.VoL + \beta_{27} Bz.AxS + \\ & + \beta_{28} Lz.Rz + \beta_{29} Lz.FrC + \beta_{30} Lz.DeN + \beta_{31} Lz.HeG + \\ & + \beta_{32} Lz.AcL + \beta_{33} Lz.VoL + \beta_{34} Lz.AxS + +\beta_{35} Rz.FrC + \\ & + \beta_{36} Rz.DeN + \beta_{37} Rz.HeG + \beta_{38} Rz.AcL + \beta_{39} Rz.VoL + \\ & + \beta_{40} Rz.AxS + \beta_{41} FrC.DeN + \beta_{42} FrC.HeG + \beta_{43} FrC.AcL + \\ & + \beta_{44} FrC.VoL + \beta_{45} FrC.AxS + \beta_{46} DeN.HeG + \beta_{47} DeN.AcL + \\ & + \beta_{48} DeN.VoL + \beta_{49} DeN.AxS + \beta_{50} HeG.AcL + \beta_{51} HeG.VoL + \\ & + \beta_{52} HeG.AxS + \beta_{53} AcL.VoL + \beta_{54} AcL.AxS + \beta_{55} Vol.AxS \end{aligned} \tag{8.7}$$

$$\begin{aligned} NB\_DBC = {} & \beta_0 + \beta_1 Fd + \beta_2 Bd + \beta_3 Ld + \beta_4 Rd + \beta_5 FrC + \beta_6 DeN + \\ & + \beta_7 HeG + \beta_8 AcL + \beta_9 VoL + \beta_{10} AxS + \beta_{11} Fd.Bd + \\ & + \beta_{12} Fd.Ld + \beta_{13} Fd.Rd + \beta_{14} Fd.FrC + \beta_{15} Fd.DeN + \\ & + \beta_{16} Fd.HeG + \beta_{17} Fd.AcL + \beta_{18} Fd.VoL + \beta_{19} Fd.AxS + \\ & + \beta_{20} Bd.Ld + \beta_{21} Bd.Rd + \beta_{22} Bd.FrC + \beta_{23} Bd.DeN + \\ & + \beta_{24} Bd.HeG + \beta_{25} Bd.AcL + \beta_{26} Bd.VoL + \beta_{27} Bd.AxS + \\ & + \beta_{28} Ld.Rd + \beta_{29} Ld.FrC + \beta_{30} Ld.DeN + \beta_{31} Ld.HeG + \\ & + \beta_{32} Ld.AcL + \beta_{33} Ld.VoL + \beta_{34} Ld.AxS + +\beta_{35} Rd.FrC + \\ & + \beta_{36} Rd.DeN + \beta_{37} Rd.HeG + \beta_{38} Rd.AcL + \beta_{39} Rd.VoL + \\ & + \beta_{40} R.AxS + \beta_{41} FrC.DeN + \beta_{42} FrC.HeG + \beta_{43} FrC.AcL + \\ & + \beta_{44} FrC.VoL + \beta_{45} FrC.AxS + \beta_{46} DeN.HeG + \beta_{47} DeN.AcL + \\ & + \beta_{48} DeN.VoL + \beta_{49} DeN.AxS + \beta_{50} HeG.AcL + \beta_{51} HeG.VoL + \\ & + \beta_{52} HeG.AxS + \beta_{53} AcL.VoL + \beta_{54} AcL.AxS + \beta_{55} Vol.AxS \end{aligned} \tag{8.8}$$

There are 55 independent variables in each model. However, the goal is to select the ones which are statistically significant and represent an important relation with the dependent variable. The model simplification approach used consists of a sequential procedure that, starting from the full model, generates models that include some of the potential independent variables, and uses an information criterion (IC) to compare the different models during the procedure.

The performance measure used to evaluate the chosen models was Pearson's correlation coefficient (Pearson's r). The value of the coefficient ranges between -1.0 (completely negatively correlated) to +1.0 (completely positively correlated). The ranges of the magnitude of r (Evans, 1996) are presented in Table 8.5.

Table 8.5: The categorisation of Pearson's r value given by Evans (1996)

Value of the Pearson's Correlation Coefficient	Strength of the Correlation
0.80 - 1.00	Very Strong
0.60 - 0.79	Strong
0.40 - 0.59	Moderate
0.20 - 0.39	Weak
0.00 - 0.19	Very Weak

8.5 Computational experiments

This section reports the results of a set of experiments run to evaluate the existing dynamic stability metrics in the CLP literature alongside the new dynamic stability metrics. The main use of these metrics will be for comparison against the dynamic stability performance indicators obtained using the StableCargo physics simulator. The algorithms for compacting the cargo arrangement and the StableCargo physics simulator were coded in C++ and were run on an Intel Core2 Duo at 2.20GHz.

8.5.1 Test instances

To the best of our knowledge there are no cargo arrangements that are widely used as test instances to evaluate dynamic stability metrics. For this reason we propose one set of 15 classes of test instances, with a total of 100 cargo arrangement solutions per class. These are solutions of the well-known instances of Bischoff and Ratcliff (1995a) and Davies and Bischoff (1999) provided by the algorithm of Parreño et al. (2008). The solutions were obtained with the enforcement of the full base support constraint for all boxes.

The instances were selected because they cover a wide range of situations, they were generated by an algorithm that was not developed to perform in terms of the M1 or M2 metrics, and the average percentage of volume occupied in the container is not close to full container occupation.

For acceleration scenarios four borderline manoeuvres of trucks were selected: the full braking deceleration, the tight cornering manoeuvre (left and right) and the standing start full acceleration. These manoeuvres generate, respectively, longitudinal and transverse acceleration. According to IMO/ILO/UNECE (2014) the arrangement of cargo must be designed to withstand forces in longitudinal and transverse directions. The applicable acceleration values for road transport in the forward longitudinal direction is $0.8\,g$ and in the transverse directions and in the backward longitudinal direction is $0.5\,g$. Each acceleration scenario is applied along the four directions (X^-, X^+, Y^+ and Y^-)

Full braking deceleration is the greatest forward acceleration that a cargo experiences during road transportation. The instance represents the full braking of a truck on a level road from 90 km/h with $0.8\,g$ braking deceleration. The tight cornering manoeuvre represents the cornering of a truck on a level road with rapid build-up in transverse acceleration up to its maximum value of $0.5\,g$ over a period of 6 seconds. These acceleration scenarios were used by Kaps (2011) in the evaluation of cargo securing regulation.

Two different sets were used to describe the physical properties of the boxes. In the first set the density of the boxes was set to 200 kg/m^3 and the friction coefficient varied in the range 0.1 to 0.8 (with intervals of 0.1), and in the second set the friction coefficient was set to 0.2 and the density varied in the range 100 kg/m^3 to 800 kg/m^3 (with intervals of 100 kg/m^3).

In total the number of simulations performed was $1500\,(arrangements) \times 2\,(accelera-tions) \times 4\,(vertices) \times 15\,(density.friction) = 180.000$.

8.5.2 Evaluation Methodology

The main goal in this paper is to develop new dynamic stability metrics that can be used within the CLP. To this end, our evaluation methodology is based on determining whether the new CLP dynamic stability performance indicators are dependent on the variables in the multiple linear regression model, and if so, the strength of that relationship. With this in mind, our evaluation methodology is as follows. Firstly, for each class of instances in the different acceleration scenarios and with different physical box properties, the correlations between the values of the dynamic performance indicators and the values of M1, M2 and %Vol are measured. Secondly, the parameters for the regression models are estimated and the models' performance evaluated.

8.5.3 Correlation between new performance indicators and literature dynamic stability metrics

To measure the correlation between the new dynamic stability performance indicators and literature metrics, we used the simulation results obtained with a fixed density value for the boxes of 200 kg/m^3 and a varying friction coefficient, and the simulation results obtained with a fixed friction coefficient of 0.2 and a varying density value for the boxes, for the acceleration scenarios in all four directions.

Table 8.6 presents the average correlation between the NFB in each class of problems and the metrics M1, M2 and %Vol. As can be observed there is a very weak correlation between the NFB and the M1 and M2 metrics and a weak correlation between the NFB and the %Vol. The maximum value for the correlation with M1 is 0.08, with M2 it is -0.05 and with the %Vol it is -0.43.

The same can be observed by analysing the average measured correlation between the NB_DBC and the literature metrics. Since a high number of NB_DBC results are zero for the $0.5g$ acceleration scenarios, the correlation was only determined for the forward $0.8g$ acceleration scenario. In this case, the maximum value for the correlation with M1 is 0.08, with M2 it is 0.05 and with the %Vol it is -0.29. For both the NFB and NB_DBC metrics the correlation with the %Vol is considerably higher than M1 and M2.

Several conclusions can de drawn from the previous results. The correlation of the proposed dynamic stability measures with M1 and M2 are very weak, and with the %Vol is weak. These results seem to indicate that the metrics used to measure dynamic stability in the literature are not up to the purpose and highlight the need of better dynamic stability metrics for the CLP.

8.5.4 Multiple regression results

The average $Num\Delta z$ and the average $HDisp$ calculated in each direction for the 100 instances of each class of instances, and the aggregate results for classes BR1–7, BR8–15 and BR1–15 are presented in Table 8.7.

The fundamental approach to model the variables NFB and NB_DBC is to perform a multiple linear regression with pairwise-interactions, as defined earlier. The exploration of the subset of reduced models with different independent variable combinations based on an IC approach was performed as implemented in the `glmulti` R package (Calcagno and Mazancourt, 2010; Calcagno, 2013). To compare the models in the subset the Akaike Information Criterion (AIC) was used. The `glmulti` function allows the implementation of an exhaustive approach to compare all candidate models or the use of a genetic algorithm to search the space of possible models. Due to the extremely large number of independent variables, the genetic algorithm method was used. The data set used in the multiple linear regression analysis is composed of the simulation results referred to in Subsection 8.5.1.

NFB Model

The results of the estimation equation for the NFB, using model (8.9), are presented in Table 8.8. As can be observed the model has a very strong regression coefficient of 0.87. The results of analysis of variance also supported the strong linear relationships in the model. The F-value for the regression was 13944.6 indicating a high level of significance (p-value <2.2e-16), rejecting the null hypothesis (H_0) that every coefficient of the independent variables in the model was zero. It must nevertheless be stated that due to the sample dimension, this result might be expected. All this evidence showed a very strong linear relationship between the independent variables and the NFB.

Table 8.6: Average correlation between the new and literature dynamic stability metrics

New Metrics	Direction	Literature Metrics	Friction Coefficient								Density (kg/m3)							
			0.1	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**100**	**200**	**300**	**400**	**500**	**600**	**700**	**800**
NFB	X^- (0.8g)	M1	0.01	0.02	0.02	0.03	0.03	0.04	0.03	0.04	0.03	0.02	0.03	0.02	0.02	0.03	0.02	0.02
		M2	0.00	0.00	0.02	0.02	0.01	−0.01	0.01	0.00	0.01	0.00	0.00	0.01	0.02	0.01	0.01	0.01
		%Vol	−0.43	−0.41	−0.38	−0.39	−0.37	−0.34	−0.33	−0.34	−0.42	−0.41	−0.40	−0.40	−0.41	−0.41	−0.39	−0.40
	Y^- (0.5g)	M1	0.04	0.00	0.03	0.01	0.03	0.04	0.05	0.06	0.00	0.00	0.01	0.01	0.00	0.01	0.01	0.01
		M2	−0.02	−0.03	−0.05	−0.03	−0.02	−0.02	−0.01	−0.01	−0.04	−0.03	−0.02	−0.03	−0.02	−0.02	−0.03	−0.03
		%Vol	−0.38	−0.32	−0.31	−0.34	−0.35	−0.35	−0.35	−0.33	−0.33	−0.32	−0.32	−0.34	−0.34	−0.34	−0.34	−0.35
	Y^+ (0.5g)	M1	0.06	0.03	0.04	0.03	0.08	0.06	0.07	0.06	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03
		M2	−0.05	−0.03	−0.04	−0.05	0.00	−0.01	0.00	0.01	−0.04	−0.03	−0.04	−0.03	−0.03	−0.03	−0.03	−0.04
		%Vol	−0.43	−0.41	−0.39	−0.35	−0.39	−0.38	−0.37	−0.36	−0.41	−0.41	−0.40	−0.41	−0.40	−0.41	−0.40	−0.41
	X^+ (0.5g)	M1	0.05	0.05	0.06	0.04	0.05	0.05	0.05	0.03	0.07	0.05	0.04	0.06	0.04	0.06	0.04	0.05
		M2	0.00	0.01	0.01	0.00	0.00	−0.02	−0.04	−0.02	0.01	0.01	0.00	0.00	−0.01	0.02	0.01	0.01
		%Vol	−0.35	−0.33	−0.31	−0.31	−0.27	−0.26	−0.23	−0.22	−0.35	−0.33	−0.32	−0.33	−0.35	−0.34	−0.35	−0.33
NB_DBC	X^- (0.8g)	M1	0.06	0.07	0.08	0.06	0.07	0.02	0.01	0.01	0.05	0.07	0.07	0.07	0.05	0.06	0.08	0.05
		M2	0.01	0.00	0.01	0.00	0.03	0.05	0.02	0.02	0.03	0.00	0.01	0.02	0.00	0.01	0.00	0.00
		%Vol	−0.24	−0.29	−0.25	−0.24	−0.23	−0.19	−0.17	−0.19	−0.29	−0.29	−0.28	−0.28	−0.26	−0.28	−0.25	−0.27

Table 8.7: Average values of the geometric independent variables

Class	$Num\Delta z_{X^-}$	$HDisp_{X^-}$	$Num\Delta z_{X^+}$	$HDisp_{X^+}$	$Num\Delta z_{Y^-}$	$HDisp_{Y^-}$	$Num\Delta z_{Y^+}$	$HDisp_{Y^+}$
BR1	0.3	281	0.0	2347	0.1	736	0.3	1848
BR2	0.6	491	0.1	2261	0.5	904	0.5	1746
BR3	0.5	793	0.2	1887	0.8	876	0.7	1417
BR4	0.8	949	0.4	1814	0.6	910	0.8	1514
BR5	0.8	1110	0.8	1908	0.9	926	0.9	1410
BR6	0.9	1191	0.9	1856	1.1	900	1.2	1468
BR7	1.2	1298	0.8	1761	0.9	1091	1.0	1526
BR8	1.1	1451	1.4	1727	1.5	1138	1.8	1533
BR9	1.3	1570	1.7	1807	1.1	1054	2.0	1502
BR10	1.7	1599	1.9	1820	1.7	1169	2.1	1507
BR11	1.5	1510	2.1	1872	1.4	1207	2.0	1385
BR12	1.8	1658	2.9	1994	1.6	1307	1.9	1397
BR13	2.8	1776	2.6	1886	1.9	1264	2.2	1414
BR14	3.1	1851	3.6	2052	2.0	1242	2.0	1305
BR15	3.2	1780	3.7	1988	2.1	1239	2.0	1311
Mean (BR 1-7)	0.7	873	0.5	1976	0.7	906	0.8	1561
Mean (BR 8-15)	2.1	1649	2.5	1893	1.7	1203	2.0	1419
Mean (BR 1-15)	1.4	1287	1.5	1932	1.2	1064	1.4	1486

$$\begin{aligned} NFB \sim Fz + Bz + Lz + Rz + AcL + VoL + FrC + HeG + \\ AxS + DeN + Fz{:}Bz + Fz{:}Lz + Bz{:}Lz + Fz{:}Rz + Lz{:}Rz + \\ Fz{:}AcL + Rz{:}AcL + Fz{:}VoL + Bz{:}VoL + Rz{:}VoL + AcL{:}VoL + \\ Fz{:}FrC + Bz{:}FrC + Rz{:}FrC + AcL{:}FrC + VoL{:}FrC + \\ Fz{:}HeG + Lz{:}HeG + AcL{:}HeG + VoL{:}HeG + FrC{:}HeG + \\ Fz{:}AxS + Lz{:}AxS + Rz{:}AxS + AcL{:}AxS + VoL{:}AxS + \\ FrC{:}AxS + HeG{:}AxS + Fz{:}DeN + FrC{:}DeN \end{aligned} \tag{8.9}$$

NB_DBC Model

The results of the estimation equation for the NB_DBC are presented in Table 8.9. The model presents a strong regression coefficient of 0.61, even considering that it is below the value found for the NFB model. With an F-value for the regression of 2523.7, the null hypothesis (H_0) that all coefficients of the independent variables in the model are zero is easily rejected, indicating a high level of significance for the regression (p-value <2.2e-16). All this evidence showed a strong linear relationship between the independent variables and the NB_DBC.

Table 8.8: Multiple regression of the NFB

Independent variable	Coefficients	Std. Error	t-statistic	p-value
Fz	1.23725	2.34E-02	52.9	<2e-16
Bz	-0.11542	2.03E-02	-5.7	1.21E-08
Lz	0.11031	5.34E-03	20.7	<2e-16
Rz	0.09568	2.11E-02	4.5	5.88E-06
AcL	1.05901	3.61E-01	2.9	3.33E-03
VoL	-1.93488	3.53E-01	-5.5	4.27E-08
FrC	8.92856	1.40E+00	6.4	1.68E-10
HeG	0.10061	3.24E-03	31.1	<2e-16
AxS	-0.42021	3.24E-02	-13.0	<2e-16
DeN	0.01048	1.46E-03	7.2	6.91E-13
Fz:Bz	0.00720	5.65E-04	12.7	<2e-16
Fz:Lz	0.00229	5.13E-04	4.5	7.78E-06
Bz:Lz	-0.00396	5.21E-04	-7.6	2.99E-14
Fz:Rz	-0.00694	5.32E-04	-13.0	<2e-16
Lz:Rz	-0.00729	5.56E-04	-13.1	<2e-16
Fz:AcL	0.34444	1.01E-02	34.2	<2e-16
Rz:AcL	0.03610	1.01E-02	3.6	3.43E-04
Fz:VoL	-0.72103	2.84E-02	-25.4	<2e-16
Bz:VoL	0.20588	2.69E-02	7.7	1.98E-14
Rz:VoL	-0.12206	2.64E-02	-4.6	3.81E-06
AcL:VoL	-2.21279	4.34E-01	-5.1	3.52E-07
Fz:FrC	-0.98740	7.83E-03	-126.0	<2e-16
Bz:FrC	-0.03703	7.27E-03	-5.1	3.51E-07
Rz:FrC	-0.03474	7.25E-03	-4.8	1.66E-06
AcL:FrC	0.86420	1.04E-01	8.3	1.14E-16
VoL:FrC	2.04254	3.22E-01	6.3	2.26E-10
Fz:HeG	-0.00084	8.47E-05	-9.9	<2e-16
Lz:HeG	-0.00050	8.26E-05	-6.1	1.09E-09
AcL:HeG	0.01134	1.34E-03	8.5	2.63E-17
VoL:HeG	-0.14740	4.09E-03	-36.1	<2e-16
FrC:HeG	-0.01296	9.68E-04	-13.4	<2e-16
Fz:AxS	0.00838	9.00E-04	9.3	1.00E-20
Lz:AxS	-0.01221	9.01E-04	-13.6	<2e-16
Rz:AxS	0.00480	8.81E-04	5.4	5.30E-08
AcL:AxS	0.11613	1.23E-02	9.4	<2e-16
VoL:AxS	0.43936	3.81E-02	11.5	<2e-16
FrC:AxS	-0.13136	8.85E-03	-14.8	<2e-16
HeG:AxS	0.00303	1.17E-04	25.8	<2e-16
Fz:DeN	0.00003	7.16E-06	4.7	2.89E-06
FrC:DeN	-0.05254	7.30E-03	-7.2	6.07E-13

Overall equation statistics:
r=0.87
F-statistic = 13944.6 , p-value <2.2e-16

$$\begin{aligned} NB\_DBC \sim\ & Fd + Bd + Ld + Rd + AcL + VoL + FrC + HeG + \\ & AxS + DeN + Fd{:}Bd + Fd{:}Ld + Bd{:}Ld + Fd{:}Rd + Bd{:}Rd + \\ & Fd{:}AcL + Bd{:}AcL + Ld{:}AcL + Fd{:}VoL + Bd{:}VoL + \\ & Ld{:}VoL + AcL{:}VoL + Fd{:}FrC + AcL{:}FrC + Fd{:}HeG + \\ & Bd{:}HeG + Ld{:}HeG + AcL{:}HeG + FrC{:}HeG + Fd{:}AxS + \\ & Bd{:}AxS + Ld{:}AxS + AcL{:}AxS + VoL{:}AxS + FrC{:}AxS + \\ & HeG{:}AxS + Bd{:}DeN + Ld{:}DeN + Rd{:}DeN + AcL{:}DeN + \\ & VoL{:}DeN + FrC{:}DeN + HeG{:}DeN \end{aligned} \tag{8.10}$$

8.5.5 Illustrative example

The following examples illustrates the dynamic stability evaluation of a cargo arrangement, with the characteristics presented in Table 8.10, for road transportation according to the IMO/ILO/UNECE (2014). The measured $Num\Delta z$ and $HDisp$ are presented in Table 8.11. By applying models (8.9) and (8.10) we are able to estimate the NFB and the NB_DBC along each of the four directions. Tables 8.12 and 8.13 present the results for the example presented. It is estimated that the NFB in the X^+ direction is two, in the X^- is one and along Y^+ and Y^- zero. Concerning the NB_DBC in the X^- direction, one box is estimated to be within the DBC. It can be therefore concluded that the cargo arrangement does not have dynamic stability.

8.6 Conclusions and future work

In this paper the dynamic stability constraint within the CLP was addressed, and two dynamic stability performance indicators that translate the real world dynamic stability constraint were proposed, the NFB and the NB_DBC. To determine the new dynamic stability performance indicators, StableCargo, a physics simulation tool, was used. Due to the computational cost of the simulation tool, two metrics that model each of the new performance indicators were developed through multiple linear regression with pairwise interaction terms. The NFB and the NB_DBC metrics present a very strong and strong linear correlation with the dynamic stability performance indicators, respectively, and are suitable to be used within a container loading algorithm. In future work we aim to incorporate the proposed dynamic stability metrics in a container loading algorithm using the new proposed metrics.

Table 8.9: Multiple regression of the NB_DBC

Independent variable	Coefficients	Std. Error	t-statistic	p-value
Fd	0.09979	8.38E-03	11.9	<2e-16
Bd	-0.09493	8.36E-03	-11.4	<2e-16
Ld	-0.07536	7.72E-03	-9.8	<2e-16
Rd	0.00182	1.02E-03	1.8	7.38E-02
AcL	5.67057	3.25E-01	17.5	<2e-16
VoL	2.35924	2.91E-01	8.1	5.56E-16
FrC	-11.61798	1.20E+00	-9.7	<2e-16
HeG	0.00898	9.49E-04	9.5	<2e-16
AxS	-0.35909	2.92E-02	-12.3	<2e-16
DeN	-0.00933	1.16E-03	-8.0	1.04E-15
Fd:Bd	-0.00175	8.51E-05	-20.6	<2e-16
Fd:Ld	-0.00181	9.80E-05	-18.5	<2e-16
Bd:Ld	0.00109	9.81E-05	11.1	<2e-16
Fd:Rd	-0.00170	7.97E-05	-21.3	<2e-16
Bd:Rd	0.00111	9.00E-05	12.3	<2e-16
Fd:AcL	0.32026	3.80E-03	84.3	<2e-16
Bd:AcL	-0.02755	3.80E-03	-7.2	4.35E-13
Ld:AcL	-0.02678	3.77E-03	-7.1	1.19E-12
Fd:VoL	-0.12920	9.98E-03	-12.9	<2e-16
Bd:VoL	0.07709	9.97E-03	7.7	1.06E-14
Ld:VoL	0.10500	9.40E-03	11.2	<2e-16
AcL:VoL	-5.48628	3.82E-01	-14.4	<2e-16
Fd:FrC	-0.22040	2.36E-03	-93.3	<2e-16
AcL:FrC	-1.69439	9.30E-02	-18.2	<2e-16
Fd:HeG	-0.00213	3.59E-05	-59.2	<2e-16
Bd:HeG	0.00047	3.59E-05	13.1	<2e-16
Ld:HeG	0.00046	3.47E-05	13.3	<2e-16
AcL:HeG	-0.02467	1.16E-03	-21.3	<2e-16
FrC:HeG	0.00789	6.79E-04	11.6	<2e-16
Fd:AxS	0.00562	5.11E-04	11.0	<2e-16
Bd:AxS	0.00947	5.10E-04	18.6	<2e-16
Ld:AxS	0.00263	5.03E-04	5.2	1.82E-07
AcL:AxS	-0.11346	1.22E-02	-9.3	1.00E-20
VoL:AxS	0.44734	3.53E-02	12.7	<2e-16
FrC:AxS	0.03521	7.54E-03	4.7	3.02E-06
HeG:AxS	0.00154	1.07E-04	14.5	<2e-16
Bd:DeN	-0.00002	2.37E-06	-10.5	<2e-16
Ld:DeN	-0.00002	2.39E-06	-6.5	6.46E-11
Rd:DeN	-0.00001	2.39E-06	-3.5	4.27E-04
AcL:DeN	0.00110	9.57E-05	11.5	<2e-16
VoL:DeN	-0.00409	2.61E-04	-15.7	<2e-16
FrC:DeN	0.06185	6.06E-03	10.2	<2e-16
HeG:DeN	-0.00001	8.26E-07	-18.0	<2e-16

Overall equation statistics:
r=0.61
F-statistic = 2523.7, p-value <2.2e-16

Table 8.10: Values for the independent variables

Variable	Value
VoL	0.8015
FrC	0.2
AcL	0.8
AxS	5.87
DeN	200
HeG	16

Table 8.11: $Num\Delta z$ and $HDisp$ values for the cargo arrangement example

	$Num\Delta$ z	$HDisp(0.8\,g)$	$HDisp(0.5\,g)$
X^+	3	5.39	0.00
X^-	1	5.14	5.14
Y^+	0	0.00	0.00
Y^-	0	0.00	0.00

Table 8.12: NFB calculation example

Model Variables	Coefficients	Acceleration direction X^+	X^-	Y^+	Y^-
Fz	1.23725	3	1	0	0
Bz	-0.11542	1	3	0	0
Lz	0.11031	0	0	1	3
Rz	0.09568	0	0	3	1
AcL	1.05901	0.50	0.80	0.50	0.50
VoL	-1.93488	0.80	0.80	0.80	0.80
FrC	8.92856	0.20	0.20	0.20	0.20
HeG	0.10061	16.00	16.00	16.00	16.00
AxS	-0.42021	5.87	5.87	2.33	2.33
DeN	0.01048	200	200	200	200
Fz:Bz	0.00720	3.00	3.00	0.00	0.00
Fz:Lz	0.00229	0.00	0.00	0.00	0.00
Bz:Lz	-0.00396	0.00	0.00	0.00	0.00
Fz:Rz	-0.00694	0.00	0.00	0.00	0.00
Lz:Rz	-0.00729	0.00	0.00	3.00	3.00
Fz:AcL	0.34444	1.50	0.80	0.00	0.00
Rz:AcL	0.03610	0.00	0.00	1.50	0.50
Fz:VoL	-0.72103	2.40	0.80	0.00	0.00
Bz:VoL	0.20588	0.80	2.40	0.00	0.00
Rz:VoL	-0.12206	0.00	0.00	2.40	0.80
AcL:VoL	-2.21279	0.40	0.64	0.40	0.40
Fz:FrC	-0.98740	0.60	0.20	0.00	0.00
Bz:FrC	-0.03703	0.20	0.60	0.00	0.00
Rz:FrC	-0.03474	0.00	0.00	0.60	0.20
AcL:FrC	0.86420	0.10	0.16	0.10	0.10
VoL:FrC	2.04254	0.16	0.16	0.16	0.16
Fz:HeG	-0.00084	48.00	16.00	0.00	0.00
Lz:HeG	-0.00050	0.00	0.00	16.00	48.00
AcL:HeG	0.01134	8.00	12.80	8.00	8.00
VoL:HeG	-0.14740	12.82	12.82	12.82	12.82
FrC:HeG	-0.01296	3.20	3.20	3.20	3.20
Fz:AxS	0.00838	17.61	5.87	0.00	0.00
Lz:AxS	-0.01221	0.00	0.00	2.33	6.99
Rz:AxS	0.00480	0.00	0.00	6.99	2.33
AcL:AxS	0.11613	2.94	4.70	1.17	1.17
VoL:AxS	0.43936	4.70	4.70	1.87	1.87
FrC:AxS	-0.13136	1.17	1.17	0.47	0.47
HeG:AxS	0.00303	93.92	93.92	37.28	37.28
Fz:DeN	0.00003	600.00	200.00	0.00	0.00
FrC:DeN	-0.05254	40.00	40.00	40.00	40.00
	NFB	2.2	1.2	0.2	0.3

Table 8.13: NB_DBC calculation example

Model		Acceleration direction			
Variables	Coefficients	X^+	X^-	Y^+	Y^-
Fd	9.98E-02	0.000	5.140	0.000	0.000
Bd	-0.09493	5.140	5.390	0.000	0.000
Ld	-0.07536	0.000	0.000	5.140	0.000
Rd	0.001817	0.000	0.000	0.000	5.140
AcL	5.670572	0.5	0.8	0.5	0.5
VoL	2.359238	0.802	0.802	0.802	0.802
FrC	-11.618	0.2	0.2	0.2	0.2
HeG	0.008985	16	16	16	16
AxS	-0.35909	5.87	5.87	2.33	2.33
DeN	-0.00933	200	200	200	200
Fd:Bd	-0.00175	0.000	27.705	0.000	0.000
Fd:Ld	-0.00181	0.000	0.000	0.000	0.000
Bd:Ld	0.001091	0.000	0.000	0.000	0.000
Fd:Rd	-0.0017	0.000	0.000	0.000	0.000
Bd:Rd	0.001106	0.000	0.000	0.000	0.000
Fd:AcL	0.32026	0.000	4.112	0.000	0.000
Bd:AcL	-0.02755	2.570	4.312	0.000	0.000
Ld:AcL	-0.02678	0.000	0.000	2.570	0.000
Fd:VoL	-0.1292	0.000	4.120	0.000	0.000
Bd:VoL	0.077089	4.120	4.320	0.000	0.000
Ld:VoL	0.104997	0.000	0.000	4.120	0.000
AcL:VoL	-5.48628	0.401	0.641	0.401	0.401
Fd:FrC	-0.2204	0.000	1.028	0.000	0.000
AcL:FrC	-1.69439	0.100	0.160	0.100	0.100
Fd:HeG	-0.00213	0.000	82.240	0.000	0.000
Bd:HeG	0.000471	82.240	86.240	0.000	0.000
Ld:HeG	0.000461	0.000	0.000	82.240	0.000
AcL:HeG	-0.02467	8.000	12.800	8.000	8.000
FrC:HeG	0.007887	3.200	3.200	3.200	3.200
Fd:AxS	0.005622	0.000	30.172	0.000	0.000
Bd:AxS	0.009467	30.172	31.639	0.000	0.000
Ld:AxS	0.002627	0.000	0.000	11.976	0.000
AcL:AxS	-0.11346	2.935	4.696	1.165	1.165
VoL:AxS	0.447343	4.705	4.705	1.867	1.867
FrC:AxS	0.035207	1.174	1.174	0.466	0.466
HeG:AxS	0.001545	93.92	93.92	37.28	37.28
Bd:DeN	-2.5E-05	1028.00	1078.00	0.00	0.00
Ld:DeN	-1.6E-05	0.00	0.00	1028.00	0.00
Rd:DeN	-8.4E-06	0.00	0.00	0.00	1028.00
AcL:DeN	0.001101	100.00	160.00	100.00	100.00
VoL:DeN	-0.00409	160.30	160.30	160.30	160.30
FrC:DeN	0.061854	40.00	40.00	40.00	40.00
HeG:DeN	-1.5E-05	3200.00	3200.00	3200.00	3200.00
	NB_DBC	-0.1	0.9	0.0	0.0

Bibliography

Abdou, G. and Arghavani, J. (1997). Interactive ILP procedures for stacking optimization for the 3D palletization problem. *International Journal of Production Research*, 35(5):1287–1304.

Abdou, G. and Elmasry, M. (1999). 3D random stacking of weakly heterogeneous palletization problems. *International Journal of Production Research*, 37(7):1505–1524.

Abdou, G. and Yang, M. (1994). A systematic approach for the three-dimensional palletization problem. *International Journal of Production Research*, 32(10):2381–2394.

Abdou, G. and Yang, M. (1995). Multi-layer palletisation of multi-size boxes. *The International Journal of Advanced Manufacturing Technology*, 10(4):292–297.

Allen, S. D., Burke, E. K., and Mareček, J. (2012). A space-indexed formulation of packing boxes into a larger box. *Operations Research Letters*, 40(1):20–24.

Almeida, J. E., Jacob, J. T. P. N., Faria, B. M., Rossetti, R. J. F., and Coelho, A. L. (2014). Serious games for the Elicitation of way-finding behaviours in emergency situations. In *9th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–7.

American Society for Testing Materials (2010). *Standard Test Methods for Mechanical-Shock fragility of Products, Using Shock Machines*. ASTM International, ASTM D3332-99 99(2010).

Araya, I. and Riff, M. C. (2014). A beam search approach to the container loading problem. *Computers and Operations Research*, 43:100–107.

Badler, N., O'Rourke, J., and Kaufman, B. (1980). Special problems in human movement simulation. *ACM SIGGRAPH Computer Graphics*, 14(3):189–197.

Balakirsky, S., Proctor, F., Kramer, T., Kolhe, P., and Christensen, H. I. (2010). Using Simulation to Assess the Effectiveness of Pallet Stacking Methods. pages 336–349.

Ballou, R. H. (1999). *Business Logistics Management: Planning, Organizing, and Controlling the Supply Chain*. Prentice-Hall International Editions. Prentice Hall, Upper Saddle River, NJ.

Bhattacharya, A. and McGlothlin, J. D. (2012). *Occupational Ergonomics: Theory and Applications, Second Edition*. Taylor & Francis Group.

Bischoff, E. E. (1991). Stability aspects of pallet loading. *OR Spektrum*, 13(4):189–197.

Bischoff, E. E. (2006). Three-dimensional packing of items with limited load bearing strength. *European Journal of Operational Research*, 168(3):952–966.

Bischoff, E. E., Janetz, F., and Ratcliff, M. S. W. (1995). Loading pallets with non-identical items. *European Journal of Operational Research*, 84(3):681–692.

Bischoff, E. E. and Ratcliff, M. S. W. (1995a). Issues in the development of approaches to container loading. *Omega*, 23(4):377–390.

Bischoff, E. E. and Ratcliff, M. S. W. (1995b). Loading Multiple Pallets. *Journal of the Operational Research Society*, 46(11):1322–1336.

Blum, C. and Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys (CSUR)*, 35(3):268–308.

Boeing, A. and Bräunl, T. (2007). Evaluation of real-time physics simulation systems. In *Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia - GRAPHITE '07*, volume 1, page 281, New York, New York, USA. ACM Press.

Bortfeldt, A. (2012). A hybrid algorithm for the capacitated vehicle routing problem with three-dimensional loading constraints. *Computers & Operations Research*, 39(9):2248–2257.

Bortfeldt, A. and Gehring, H. (1999). Two metaheuristics for strip packing problems. In *Erscheint in: Proceedingsband der 5th International Conference of the Decision Sciences Institute, Athen*.

Bortfeldt, A. and Gehring, H. (2001). A hybrid genetic algorithm for the container loading problem. *European Journal of Operational Research*, 131(1):143–161.

Bortfeldt, A., Gehring, H., and Mack, D. (2003). A parallel tabu search algorithm for solving the container loading problem. *Parallel Computing*, 29(5):641–662.

Bortfeldt, A. and Wäscher, G. (2013). Constraints in container loading-A state-of-the-art review. *European Journal of Operational Research*, 229(1):1–20.

Brandenburg, R. K. and Lee, J. J. L. (1985). *Fundamentals of Packaging Dynamics*. MTS Systems Corporation, Minneapolis, MN.

Braun, a., Musse, S., Oliveira, L. D., and Bodmann, B. (2003). Modeling individual behaviors in crowd simulation. *Proceedings 11th IEEE International Workshop on Program Comprehension*, (6).

Calcagno, V. (2013). glmulti: Model selection and multimodel inference made easy (Version 1.0.7) [Computer software]. Available from http://cran.r-project.org/web/packages/glmulti/index.html.

Calcagno, V. and Mazancourt, C. D. (2010). glmulti : An R Package for Easy Automated Model Selection with (Generalized) Linear Models. *Journal of statistical software*, 34(12):1–29.

Candea, C., Hu, H., Iocchi, L., Nardi, D., and Piaggio, M. (2001). Coordination in multi-agent RoboCup teams. In *Robotics and Autonomous Systems*, volume 36, pages 67–86.

Carpenter, H. and Dowsland, W. B. (1985). Practical Considerations of the Pallet-Loading Problem. *Journal of the Operational Research Society*, 36(6):489–497.

Ceschia, S. and Schaerf, A. (2011). Local search for a multi-drop multi-container loading problem. *Journal of Heuristics*.

Chan, F. T., Bhagwat, R., Kumar, N., Tiwari, M., and Lam, P. (2006). Development of a decision support system for air-cargo pallets loading problem: A case study. *Expert Systems with Applications*, 31(3):472–485.

Chand, D. R. and Kapur, S. S. (1970). An Algorithm for Convex Polytopes. *Journal of the ACM*, 17(1):78–86.

Chen, C., Lee, S., and Shen, Q. (1995). An analytical model for the container loading problem. *European Journal of Operational Research*, 80(1):68–76.

Christensen, S. r. G. and Rousøe, D. M. (2009). Container loading with multi-drop constraints. *International Transactions in Operational Research*, 16(6):727–743.

Corporation, N. (2004). PhysX engine [Computer software]. Available from https://developer.nvidia.com/gameworks-physx-overview.

Coumans, E. (2013). Bullet Physics Library (Version 2.82) [Computer software]. Available from http://bulletphysics.org/.

Crainic, T. G., Perboli, G., and Tadei, R. (2008). Extreme Point-Based Heuristics for Three-Dimensional Bin Packing. *INFORMS Journal on Computing*, 20(3):368–384.

Davies, A. and Bischoff, E. E. (1999). Weight distribution considerations in container loading. *European Journal of Operational Research*, 114(3):509–527.

de Araújo, O. C. B. and Armentano, V. A. (2007). A multi-start random constructive heuristic for the container loading problem. *Pesquisa Operacional*, 27(2):311–331.

de Castro Silva, J. L., Soma, N. Y., and Maculan, N. (2003). A greedy search for the three-dimensional bin packing problem: the packing static stability case. *International Transactions in Operational Research*, 10(2):141–153.

Dibakar, S. and Mruthyunjaya, T. (1999). A computational geometry approach for determination of boundary of workspaces of planar manipulators with arbitrary topology. *Mechanism and Machine Theory*, 34(1):149–169.

Dyckhoff, H. (1990). A typology of cutting and packing problems. *European Journal of Operational Research*, 44(2):145–159.

EC (2010). *European Best Practice Guidelines on Cargo Securing for Road Transport*. European Commision, Brussels.

EC (2011). White Paper on Transport. Technical report, European Commission.

EC (2012). Cargo securing and abnormal loads.

ECS (2010). *EN 12195: Load restraint assemblies on road vehicles – Safety*. European Committee for Standardization, Brussels.

Edelsbrunner, H. and Maurer, H. A. (1981). On the intersection of orthogonal objects. *Information Processing Letters*, 13(4/5):177–181.

Egeblad, J., Garavelli, C., Lisi, S., and Pisinger, D. (2010). Heuristics for container loading of furniture. *European Journal of Operational Research*, 200(3):881–892.

Egeblad, J. and Pisinger, D. (2009). Heuristic approaches for the two- and three-dimensional knapsack packing problem. *Computers & Operations Research*, 36(4):1026–1049.

Eley, M. (2002). Solving container loading problems by block arrangement. *European Journal of Operational Research*, 141(2):393–409.

Erleben, K. (2002). Module based design for rigid body simulators. Technical report, University of Copenhagen.

Evans, J. D. (1996). *Straightforward Statistics for the Behavioral Sciences*. Brooks/Cole Publishing Company, Pacific Grove/ California.

Fanslau, T. and Bortfeldt, A. (2010). A Tree Search Algorithm for Solving the Container Loading Problem. *INFORMS Journal on Computing*, 22(2):222–235.

Fasano, G. (1999). *Cargo Analytical Integration in Space Engineering: A Three-dimensional Packing Model*. Palgrave Macmillan.

Fekete, S. and Schepers, J. (1997). A new exact algorithm for general orthogonal d-dimensional knapsack problems. *Algorithms—ESA'97*, 1284:144–156.

Fekete, S. P., Schepers, J., and van der Veen, J. C. (2007). An Exact Algorithm for Higher-Dimensional Orthogonal Packing. *Operations Research*, 55(3):569–587.

Fiedler, R. M. (2009). Shock. In Yam, K. L., editor, *The Wiley Encyclopedia of Packaging Technology*, pages 1107–1111. John Wiley & Sons, Inc.

Fuellerer, G., Doerner, K. F., Hartl, R. F., and Iori, M. (2010). Metaheuristics for vehicle routing problems with three-dimensional loading constraints. *European Journal of Operational Research*, 201(3):751–759.

GDV (2003). *Container Handbook*. German Insurance Association.

Gehring, H. and Bortfeldt, A. (1997). A Genetic Algorithm for Solving the Container Loading Problem. *International Transactions in Operational Research*, 4(5-6):401–418.

Gehring, H. and Bortfeldt, A. (2002). A Parallel Genetic Algorithm for Solving the Container Loading Problem. *International Transactions in Operational Research*, 9(4):497–511.

Gendreau, M., Iori, M., Laporte, G., and Martello, S. (2006). A Tabu Search Algorithm for a Routing and Container Loading Problem. *Transportation Science*, 40(3):342–350.

George, J. and Robinson, D. (1980). A heuristic for packing boxes into a container. *Computers & Operations Research*, 7(3):147–156.

Gillespie, T. D. (1992). *Fundamentals of Vehicle Dynamics*. Society of Automotive Engineers, Warrendale, PA.

Gonçalves, J. F. and Resende, M. G. C. (2011). Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics*, 17:487–525.

Gonçalves, J. F. and Resende, M. G. C. (2012). A parallel multi-population biased random-key genetic algorithm for a container loading problem. *Computers and Operations Research*, 39:179–190.

Gonçalves, J. F. and Resende, M. G. C. (2013). A biased random key genetic algorithm for 2D and 3D bin packing problems. *International Journal of Production Economics*, 145(2):500–510.

Haessler, R. W. and Brian Talbot, F. (1990). Load planning for shipments of low density products. *European Journal of Operational Research*, 44(2):289–299.

Haines, E. (1994). Point in Polygon Strategies. In Heckbert, P. S., editor, *Graphics Gems 4*, The Graphics Gems Series, pages 24–46. Acad. Press.

Hellström, B. D. and Saghir, M. (2007). Packaging and logistics interactions in retail supply chains. *Packaging Technology and Science*, 20(May/June 2007):197–216.

Hemminki, J., Leipala, T., and Nevalainen, O. (1998). On-line packing with boxes of different sizes. *International journal of Production Research*, 36(8):2225–2245.

Hibbeler, R. C. (1994). *Mechanics of materials*. Prentice Hall, Englewood Cliffs, NJ, 2nd edition.

Hibbeler, R. C. (2010a). *Engineering Mechanics: Dynamics*. MasteringEngineering Series. Prentice Hall, Upper Saddle River, NJ, 12 edition.

Hibbeler, R. C. (2010b). *Engineering Mechanics: Statics*. Engineering Mechanics. Prentice Hall, Upper Saddle River, NJ.

Hifi, M. (2002). Approximate algorithms for the container loading problem. *International Transactions in Operational Research*, 9(6):747–774.

Hummel, J., Wolff, R., Stein, T., Gerndt, A., and Kuhlen, T. (2012). An evaluation of open source physics engines for use in virtual reality assembly simulations. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 7432 LNCS, pages 346–357.

IMO/ILO/UNECE (2014). *IMO/ILO/UNECE Code of Practice for Packing of Cargo Transport Units*, volume 4. International Maritime Organization.

Iori, M. and Martello, S. (2010). Routing problems with loading constraints. *TOP*, 18(1):4–27.

Ivancic, N., Mathur, K., and Mohanty, B. B. (1989). An integer programming based heuristic approach to the three-dimensional packing problem. *Journal of Manufacturing and Operations Management*, 2:268–298.

Jarvis, R. A. (1973). On the Identification of the Convex Hull of a Finite Set of Points in the Plane. *Information Processing Letters*, 2(1):18–21.

Jerez, J. and Suero, A. (2003). Newton Game Dynamics [Computer software]. Available from http://newtondynamics.com.

Jin, Z., Ito, T., and Ohno, K. (2003). The Three-Dimensional Bin Packing Problem and Its Practical Algorithm. *JSME International Journal Series C*, 46(1):60–66.

Jin, Z., Ohno, K., and Du, J. (2004). An efficient approach for the three-dimensional container packing problem with practical constraints. *Asia-Pacific Journal of Operational Research*, 21(03):279–295.

Jones, M. T. (2011). Open source physics engines Building believable worlds with open source. Technical report, IBM.

Junqueira, L., Morabito, R., and Sato Yamashita, D. (2012a). MIP-based approaches for the container loading problem with multi-drop constraints. *Annals of Operations Research*, 199(1):51–75.

Junqueira, L., Morabito, R., and Sato Yamashita, D. (2012b). Three-dimensional container loading models with cargo stability and load bearing constraints. *Computers & Operations Research*, 39(1):74–85.

Junqueira, L., Oliveira, J. F., Carravilla, M. A., and Morabito, R. (2013). An optimization model for the vehicle routing problem with practical three-dimensional loading constraints. *International Transactions in Operational Research*, 20(5):645–666.

Kaps, H. (2011). Securing cargo in road transport – Who knows the truth ? Technical Report August, Gesamtverband der Deutschen Versicherungswirtschaft.

Koonce, J. M. and Bramble Jr., W. J. (1998). Personal Computer-Based Flight Training Devices. *The International Journal of Aviation Psychology*, 8(3):277–292.

Lai, K. and Chan, J. (1997). Developing a simulated annealing algorithm for the cutting stock problem. *Computers & industrial engineering*, 32(1):115–127.

LaValle, S. M. (2006). *Planning Algorithms*. Cambridge University Press.

Levinson, M. (2008). *The Box: How the Shipping Container Made the World Smaller and the World Economy Bigger*. Princeton University Press.

Lim, a., Rodrigues, B., and Wang, Y. (2003). A multi-faced buildup algorithm for three-dimensional packing problems. *Omega*, 31(6):471–481.

Lin, J.-L., Chang, C.-H., and Yang, J.-Y. (2006). A Study of Optimal System for Multiple-Constraint Multiple-Container Packing Problems. In Ali, M. and Dapoigny, R., editors, *Advances in Applied Artificial Intelligence*, volume 4031 of *Lecture Notes in Computer Science*, pages 1200–1210. Springer Berlin / Heidelberg.

Liu, F.-H. F. and Hsiao, C.-J. (1997). A Three-Dimensional Pallet Loading Method for Single-Size Boxes. *The Journal of the Operational Research Society*, 48(7):726.

Liu, J., Yue, Y., Dong, Z., Maple, C., and Keech, M. (2011a). A novel hybrid tabu search approach to container loading. *Computers & Operations Research*, 38(4):797–807.

Liu, W.-Y., Lin, C.-C., and Yu, C.-S. (2011b). On the three-dimensional container packing problem under home delivery service. *Asia-Pacific Journal of Operational Research*, 28(05):601–621.

Loh, T. H. and Nee, A. Y. C. (1992). A packing algorithm for hexahedral boxes. In *In: Proceedings of the Conference of Industrial Automation*, pages 115–126.

Mack, D., Bortfeldt, A., and Gehring, H. (2004). A parallel hybrid local search algorithm for the container loading problem. *International Transactions in Operational Research*, 11:511–533.

Martello, S., Pisinger, D., and Vigo, D. (2000). The Three-Dimensional Bin Packing Problem. *Operations Research*, 48(2):256–267.

Morabito, R. and Arenales, M. (1994). An AND/OR-graph Approach to the Container Loading Problem. *International Transactions in Operational Research*, 1(1):59–73.

Moura, A. and Oliveira, J. F. (2005). A GRASP Approach to the Container-Loading Problem. *IEEE Intelligent Systems*, 20(4):50–57.

Moura, A. and Oliveira, J. F. (2008). An integrated approach to the vehicle routing and container loading problems. *OR Spectrum*, 31(4):775–800.

Mustafee, N. and Bischoff, E. E. (2013). Analysing trade-offs in container loading: combining load plan construction heuristics with agent-based simulation. *International Transactions in Operational Research*, 20(4):471–491.

Newton, R. (1968). Fragility assessment theory and test procedure. Technical report, Monterey Research Laboratory, Monterey, CA, USA.

Ngoi, B. K. A., Tay, M. L., and Chua, E. S. (1994). Applying spatial representation techniques to the container packing problem. *International Journal of Production Research*, 32(1):111–123.

Ngoi, B. K. A. and Whybrew, K. (1993). A fast spatial representation method (applied to fixture design). *The International Journal of Advanced Manufacturing Technology*, 8(2):71–77.

Nordbeck, S. and Rystedt, B. (1967). Computer cartography point-in-polygon programs. *BIT Numerical Mathematics*, 7(1):39–64.

Padberg, M. (2000). Packing small boxes into a big box. *Mathematical Methods of Operations Research (ZOR)*, 52(1):1–21.

Parreño, F., Alvarez-Valdes, R., Oliveira, J. F., and Tamarit, J. M. (2010). Neighborhood structures for the container loading problem: a VNS implementation. *Journal of Heuristics*, 16(1):1–22.

Parreño, F., Alvarez-Valdes, R., Tamarit, J. M., and Oliveira, J. F. (2008). A Maximal-Space Algorithm for the Container Loading Problem. *INFORMS Journal on Computing*, 20(3):412–422.

Pepper, C., Balakirsky, S., and Scrapper, C. (2007). Robot simulation physics validation. In *Proceedings of the 2007 Workshop on Performance Metrics for Intelligent Systems - PerMIS '07*, pages 97–104, New York, New York, USA. ACM Press.

Pinciroli, C., Trianni, V., O'Grady, R., Pini, G., Brutschy, A., Brambilla, M., Mathews, N., Ferrante, E., Di Caro, G., Ducatelle, F., Stirling, T., Gutiérrez, A., Gambardella, L. M., and Dorigo, M. (2011). ARGoS: A modular, multi-engine simulator for heterogeneous swarm robotics. In *IEEE International Conference on Intelligent Robots and Systems*, pages 5027–5034.

Pisinger, D. (2002). Heuristics for the container loading problem. *European Journal of Operational Research*, 141(2):382–392.

Rademacher, P. (2006). GLUI User Interface Library.

Ramos, A., Jacob, J. a., Justo, J., Oliveira, J. F., Rodrigues, R., and Gomes, A. M. (2014a). A physics simulation tool for the container loading problem. In *Proceedings of the 26th European Modeling and Simulation Symposium*.

Ramos, A. G. a., Oliveira, J. F., and Lopes, M. (2014b). A physical packing sequence algorithm for the container loading problem with static mechanical equilibrium conditions. *International Transactions in Operational Research*, 00:1–24.

Ratcliff, M. S. W. and Bischoff, E. E. (1998). Allowing for weight considerations in container loading. *OR Spektrum*, 20(1):65–71.

Ren, J., Tian, Y., and Sawaragi, T. (2011). A tree search method for the container loading problem with shipment priority. *European Journal of Operational Research*, 214(3):526–535.

Rodrigues, R., Marques, T., and Jacob, J. a. (2012). CGFLib - A library for Computer Graphics @ {FEUP} (Version 2.3) [Computer software]. Available from http://bit.ly/rrcgflib.

Rothlauf, F. (2011). *Design of Modern Heuristics: Principles and Application*. Natural Computing Series. Springer.

Scheithauer, G. and Terno, J. (1997). A heuristic approach for solving the multi-pallet packing problem. In Mukhacheva, E., editor, *Decisionb Making under Condition of Uncertainty*, pages 140–154.

Seugling, A. and Rolin, M. (2006). *Evaluation of physics engines and implementation of a physics module in a 3d-authoring tool*. PhD thesis, Umea University.

Shimrat, M. (1962). Algorithm 112: Position of point relative to polygon. *Commun. ACM*, 5(8):434—-.

Shreiner, D. and The Khronos OpenGL ARB Working Group (2009). *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Versions 3.0 and 3.1*. Pearson Education.

Tao, Y. and Wang, F. (2013). An effective tabu search approach with improved loading algorithms for the 3L-CVRP. *Computers & Operations Research*, pages 1–14.

Tarantilis, C., Zachariadis, E., and Kiranoudis, C. (2009). A Hybrid Metaheuristic Algorithm for the Integrated Vehicle Routing and Three-Dimensional Container-Loading Problem. *IEEE Transactions on Intelligent Transportation Systems*, 10(2):255–271.

Techanitisawad, A. and Tangwiwatwong, P. (2004). A GA-based Heuristic for the Interrelated Container Selection Loading Problems. *IEMS*, 3(1):22–37.

Terno, J., Scheithauer, G., Sommerweiß, U., and Riehme, J. (2000). An efficient approach for the multi-pallet loading problem. *European Journal of Operational Research*, 123(2):372–381.

UK P&I Club (2000). Container matters. *LP News*, pages 1–24.

Vukobratović, M. and Borovac, B. (2004). Zero-Moment Point — Thirty Five Years of Its Life. *International Journal of Humanoid Robotics*, 01(01):157–173.

Wäscher, G., Hauß ner, H., and Schumann, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183(3):1109–1130.

Williams, B. (2011). *Scenario-Based Training with X-Plane and Microsoft Flight Simulator: Using PC-Based Flight Simulations Based on FAA-Industry Training Standards*. ITPro collection. Wiley.

Yeung, L. and Tang, W. (2005). A Hybrid Genetic Approach for Container Loading in Logistics Industry. *IEEE Transactions on Industrial Electronics*, 52(2):617–627.

Young, D. E. (2009). Distribution Hazard Measurement. In Yam, K. L., editor, *The Wiley Encyclopedia of Packaging Technology*, chapter D, pages 365–368. John Wiley & Sons, Inc.

Zhang, D., Peng, Y., and Leung, S. C. (2012). A heuristic block-loading algorithm based on multi-layer search for the container loading problem. *Computers & Operations Research*, 39(10):2267–2276.

Zhu, W. and Lim, A. (2012). A new iterative-doubling Greedy–Lookahead algorithm for the single container loading problem. *European Journal of Operational Research*, 222(3):408–417.

Appendix A

Publications on the container loading problem

Table A.1: Characteristics in publications on the container loading problem that address stability

N.º	Publication	Problem		Assortment Large Objects		Type of Large Object		Assortment of Small Items				Type of Heuristics Method			Metaheurístics							Test Instances										
		Type	Extensions	One	Several	Container	Pallet	Identical	Weakly Heterogenious	Strongly Heterogenious	Exact Method	Constructive	Metaheuristics	Tree Search	Genetic Algorithm	Simulated Annealing	GRASP	Ant Colony Optimization	Guided Local Search	VNS	Tabu Seach	700 Bishoff and Ratcliff (1995)	800 Davis and Bishof (1998)	100 Davis and Bishof (1998)	15 Loh & Nee (1992)	Beasly (1990)	27 Gendreau el al. (2006)	47 Ivanic et .al. (1989)	Martelo at al. (2000)	Other	Authors	None
1	Abdou and Yang (1994)	SKP		r			x			r		x																				x
2	Abdou and Yang (1995)	SKP		r			x			r	x																			x		
3	Abdou and Arghavani (1997)	SKP		r			x			r	x																			x		
4	Abdou and Elmasry (1999)	SLOPP	Online	r			x		r			x																			x	
5	Araya and Riff (2014)	IPP		r			x	r						x										x								
		SKP		r			x			r				x									x									
		SLOPP		r			x		r					x								x										
6	Balakirsky et al. (2010)	SKP		r			x			r																				x		
		SLOPP		r			x		r																					x		
7	Bischoff (1991)	IIPP		r			x	r				x																			x	
8	Bischoff (2006)	SLOPP		r		x			r			x										x									x	
9	Bischoff et al. (1995)	SLOPP		r			x		r			x																			x	
10	Bischoff and Ratcliff (1995a)	SLOPP		r		x			r			x										x			x							
		SSSCSP			r (i)	x			r			x										x						x				
11	Bischoff and Ratcliff (1995b):	SSSCSP			r (i)		x		r			x																		x		
12	Bortfeldt (2012)	SBSBPP	CVRP		r (i)	x				r				x													x			x		
13	Bortfeldt and Gehring (1999)	ODP		r		x			r	r			x		x							x			x					x		
				r		x			r	r			x								x	x			x					x		
14	Bortfeldt and Gehring (2001)	SKP		r		x				r			x		x							x	x		x					x		
15	Bortfeldt et al. (2003)	SLOPP		r		x			r				x								x	x			x							
16	Carpenter and Dowsland (1985)	IIPP		r			x	r				x																			x	
17	Ceschia and Schaerf (2011)	MHKP	Multidrop		r (s)	x				r			x			x					x	x	x					x			x	
		MBSBPP	Multidrop		r(w)	x				r			x			x					x	x	x					x			x	
18	Chan et al. (2006)	MSSCSP	Multidrop		irr(w)		x		r			x																			x	
19	Christensen and Rousøe (2009)	SLOPP		r		x			r					x								x									x	
20	Davies and Bischoff (1999)	SLOPP		r		x			r			x										x	x	x								
21	de Araujo and Armentano (2007)	SLOPP		r		x			r			x										x								x		
22	de Castro Silva et al. (2003)	SBSBPP			r(i)	x				r		x																	x	x		
23	Egeblad el al. (2010)	SKP		r		x				irr				x																	x	
24	Eley (2002)	SLOPP		r		x			r			x		x								x			x							
		SSSCSP			r(i)	x			r			x		x														x			x	

r - rectangular shape; irr - irregular shape; (i) - identical; (w) - weakly heterogeneous; (s) - strongly heterogeneous

Table A.1: Characteristics in publications on the container loading that address stability(Continued)

		Problem		Assortment Large Objects		Type of Large Object		Assortment of Small Items				Type of Heuristics Method			Metaheurístics								Test Instances										
N.º	Publication	Type	Extensions	One	Several	Container	Pallet	Identical	Weakly Heterogenious	Strongly Heterogenious	Exact Method	Constructive	Metaheuristics	Tree Search	Genetic Algorithm	Simulated Annealing	GRASP	Ant Colony Optimization	Guided Local Search	VNS	Tabu Seach	700 Bishoff and Ratcliff (1995)	800 Davis and Bishof (1998)	100 Davis and Bishof (1998)	15 Loh & Nee (1992)	Beasly (1990)	27 Gendreau el al. (2006)	47 Ivanic et .al. (1989)	Martelo at al. (2000)	Other	Authors	None
25	Fanslau and Bortfeldt (2010)	IPP		r		x		r						x										x								
		SKP		r		x				r				x									x									
		SLOPP		r		x			r					x								x										
26	Fuellerer et al. (2010)	SBSBPP	VRP		r(i)	x				r			x					x										x				
27	Gehring and Bortfeldt (1997)	SKP		r		x				r			x		x								x			x						x	
28	Gehring and Bortfeldt (2002)	SKP		r		x				r			x		x								x	x								x	
29	Gendreau et al. (2006)	SBSBPP			r(i)	x				r			x									x						x				x	
30	Goncalves and Resende (2012)	SKP		r		x				r			x		x									x									
		SLOPP		r		x			r				x		x								x										
31	Haessler and Talbot (1990):	SKP		r		x			r			x																				x	
32	Hemminki et al. (1998)	SBSBPP	Online		r(i)		x			r		x																				x	
33	Hifi (2002)	SLOPP		r		x			r			x		x																	x	x	
34	Iori and Martello (2010):	SBSBPP	VRP		r(i)	x				r																					x		
35	Jin et al. (2003)	SBSBPP			r(i)	x				r	x		x									x								x			
36	Jin et al. (2004)	SLOPP		r		x			r				x			x											x						
37	Junqueira et al. (2012b)	SLOPP		r		x			r		x																					x	
38	Junqueira et al. (2012a)	SLOPP		r		x			x		x																				x	x	
39	Junqueira et al. (2013)	SSSCSP	VRP		r(i)	x			x		x																					x	
40	Lin et al. (2006):	RBPP			r(s)	x				r			x		x																	x	
41	Liu and Hsiao (1997)	IIPP		r			x	x				x																				x	
42	Liu, W. et al. (2011)	SKP		r		x				r		x															x						
43	Liu, J. et al. (2011)	SKP		r		x				r			x									x		x								x	
		SLOPP		r		x			r				x									x	x									x	
44	Mack et al. (2004)	SLOPP		r		x			r				x			x						x	x										
45	Morabito and Arenales (1994):	SLOPP		r		x			r			x		x																		x	
46	Moura and Oliveira (2005)	SKP		r		x				r			x				x						x	x		x							
		SLOPP		r		x			r				x				x						x	x		x							
47	Moura and Oliveira (2009)	SSSCSP	VRPTW		r(i)	x			r			x	x				x															x	
48	Mustafee and Bischoff (2013)	SLOPP		r		x			r			x																				x	
49	Ngoi et al. (1994)	SKP		r		x				r		x														x							

r - rectangular shape; irr - irregular shape; (i) - identical; (w) - weakly heterogeneous; (s) - strongly heterogeneous

Table A.1: Characteristics in publications on the container loading that address stability(Continued)

		Problem		Assortment Large Objects		Type of Large Object		Assortment of Small Items				Type of Heuristics Method			Metaheurísticas							Test Instances									
N.º	Publication	Type	Extensions	One	Several	Container	Pallet	Identical	Weakly Heterogenious	Strongly Heterogenious	Exact Method	Constructive	Metaheuristics	Tree Search	Genetic Algorithm	Simulated Annealing	GRASP	Ant Colony Optimization	Guided Local Search	VNS	Tabu Seach	700 Bishoff and Ratcliff (1995)	800 Davis and Bishof (1998)	100 Davis and Bishof (1998)	15 Loh & Nee (1992)	Beasly (1990)	27 Gendreau el al. (2006)	47 Ivanic et. al. (1989)	Martelo at al. (2000)	Other	Authors	None
50	Parreño at al. (2008)	SKP		r		x				r			x				x						x								x	
		SLOPP		r		x			r				x				x					x										
51	Parreño et al. (2010b)	SKP		r		x				r			x							x				x								
		SLOPP		r		x			r				x							x		x										
52	Ratcliff and Bischoff (1998)	SLOPP		r		x			r			x											x			x						
53	Ren et al. (2011)	SLOPP		r		x			r					x									x			x						
54	Scheithauer and Terno (1997)	SSSCSP			r(i)		x		r			x																			x	
55	Tao and Wang (2013)	SBSBPP	VRP		r(i)	x				r												x						x				
56	Tarantilis et al. (2009)	SBSBPP	VRP		r(i)	x				r			x							x		x						x			x	
57	Techanitisawad and Tangwiwatwong (2004)	SKP		r		x				r			x		x											x				x		
		MBSBPP			r(w)	x			r				x		x											x				x		
58	Terno et al. (2000)	SSSCSP			r(i)		x		r			x																			x	
59	Yeung and Tang (2005)	ODP		r		x				x			x		x																x	
60	Zhang et al. (2012)	IPP		r		x		r						x											x							
		SKP		r		x				r				x										x								
		SLOPP		r		x			r					x									x									
61	Zhu and Lim (2012)	IPP		r		x		r						x											x							
		SKP		r		x				r				x										x								
		SLOPP		r		x			r					x									x									

r - rectangular shape; irr - irregular shape; (i) - identical; (w) - weakly heterogeneous; (s) - strongly heterogeneous

Table A.2: Constraints and filling strategies in publications on the container loading problem that address stability

		Problem		Safety Constraints						Logistics constraints					Stability						Box arrangement Strategy				Spatial Representation				Loading Patterns				
N.º	Publication	Type	Extensions	Weight limit	Weight Distribution	Orientation	Stacking	Positioning	Stability	Complete shipments	Loading Priorities	Allocation	Positioning	Pattern Complexity	Vertical	Horizontal	Full	Partial	Mechanical	Stability Metrics	In	1D	2D	3D	Cartesian	Ngoi	Corner points	Empty spaces	Wall Building	Stack Building	Horizontal Layer Building	Block Building	Guilhotine Cutting
1	Abdou and Yang (1994)	SKP				x			x						x	x	x			x				x								x	
2	Abdou and Yang (1995)	SKP				x			x						x		x						x									x		
3	Abdou and Arghavani (1997)	SKP				x			x						x		x							x						x				
4	Abdou and Elmasry (1999)	SLOPP	Online			x			x						x		x				x							x						
5	Araya and Riff (2014)	IPP				x			x						x		x							x				x						
		SKP				x			x						x		x							x				x						
		SLOPP				x			x						x		x							x				x						
6	Balakirsky et al. (2010)	SKP		x	x		x	x	x				x		x					x													
		SLOPP		x	x		x	x	x				x		x					x													
7	Bischoff (1991)	IIPP							x						x			x		x			x									x	
8	Bischoff (2006)	SLOPP				x	x		x						x		x				x					x								
9	Bischoff et al. (1995)	SLOPP				x			x						x		x						x					x				x		
10	Bischoff and Ratcliff (1995a)	SLOPP				x		x	x				x		x	x				x			x					x				x		
		SSSCSP				x		x	x				x		x	x	x				x							x		x				
11	Bischoff and Ratcliff (1995b):	SSSCSP							x						x		x						x					x			x		
12	Bortfeldt (2012)	SBSBPP	CVRP	x		x	x	x	x			x	x		x			x			x						x						
13	Bortfeldt and Gehring (1999)	ODP				x			x						x		x				x							x	x					
						x			x						x		x							x				x						
14	Bortfeldt and Gehring (2001)	SKP		x	x	x	x		x					x	x	x	x			x	x							x	x					
15	Bortfeldt et al. (2003)	SLOPP				x			x						x		x	x		x				x				x				x		
16	Carpenter and Dowsland (1985)	IIPP							x					x				x					x									x		
17	Ceschia and Schaerf (2011)	MHKP	Multidrop	x		x	x	x	x				x		x		x				x								x					
		MBSBPP	Multidrop	x		x	x	x	x				x		x		x				x								x					
18	Chan et al. (2006)	MSSCSP	Multidrop	x	x	x		x	x			x	x			x								x				x				x		
19	Christensen and Rousøe (2009)	SLOPP				x	x	x	x				x		x		x	x			x							x						
20	Davies and Bischoff (1999)	SLOPP			x	x			x						x	x	x			x				x				x	x			x		
21	de Araujo and Armentano (2007)	SLOPP				x			x						x	x	x			x				x		x						x		
22	de Castro Silva et al. (2003)	SBSBPP				x			x						x				x		x						x							
23	Egeblad el al. (2010)	SKP		x	x	x	x	x	x				x	x	x				x	x	x			x					x			x		
24	Eley (2002)	SLOPP			x	x			x						x	x	x			x				x				x				x		
		SSSCSP			x	x			x						x	x	x			x			x					x				x		
25	Fanslau and Bortfeldt (2010)	IPP				x			x					x	x		x	x		x				x				x				x		
		SKP				x			x					x	x		x	x		x				x				x				x		
		SLOPP				x			x					x	x		x	x		x				x				x				x		
26	Fuellerer et al. (2010)	SBSBPP	VRP	x		x	x	x	x			x	x		x			x			x								x					
27	Gehring and Bortfeldt (1997)	SKP		x	x	x	x		x						x			x						x				x		x				
28	Gehring and Bortfeldt (2002)	SKP		x		x	x		x					x	x		x							x				x	x					
29	Gendreau et al. (2006)	SBSBPP		x		x	x	x	x			x	x		x			x			x							x						
30	Goncalves and Resende (2012)	SKP				x			x						x		x							x				x						
		SLOPP				x			x						x		x							x				x						

Table A.2: Constraints and filling strategies in publications on the container loading problem that address stability (Continued)

		Problem		Safety Constraints						Logistics constraints					Stability						Box arrangement Strategy				Spatial Representation				Loading Patterns				
N.º	Publication	Type	Extensions	Weight limit	Weight Distribution	Orientation	Stacking	Positioning	Stability	Complete shipments	Loading Priorities	Allocation	Positioning	Pattern Complexity	Vertical	Horizontal	Full	Partial	Mechanical	Stability Metrics	In	1D	2D	3D	Cartesian	Ngoi	Corner points	Empty spaces	Wall Building	Stack Building	Horizontal Layer Building	Block Building	Guilhotine Cutting
31	Haessler and Talbot (1990):	SKP			x	x		x	x				x	x	x			x						x					x				
32	Hemminki et al. (1998)	SBSBPP	Online			x			x						x		x	x			x						x	x					
33	Hifi (2002)	SLOPP				x			x					x	x		x													x			
34	Iori and Martello (2010):	SBSBPP	VRP	x		x	x	x	x			x	x		x			x															
35	Jin et al. (2003)	SBSBPP				x			x						x		x				x							x					
36	Jin et al. (2004)	SLOPP				x		x	x				x		x		x				x							x					
37	Junqueira et al. (2012b)	SLOPP				x	x		x						x	x	x	x															
38	Junqueira et al. (2012a)	SLOPP						x	x				x		x		x																
39	Junqueira et al. (2013)	SSSCSP	VRP				x	x	x				x		x		x																
40	Lin et al. (2006):	RBPP				x	x	x	x				x		x				x												x		
41	Liu and Hsiao (1997)	IIPP							x						x			x		x			x								x		
42	Liu, W. et al. (2011)	SKP				x		x	x				x		x		x				x							x					
43	Liu, J. et al. (2011)	SKP		x	x	x			x			x			x	x	x			x				x				x				x	
		SLOPP		x	x	x			x			x			x	x	x			x				x				x				x	
44	Mack et al. (2004)	SLOPP				x			x						x				x					x				x				x	
45	Morabito and Arenales (1994):	SLOPP				x			x					x	x		x								x				x		x	x		x
46	Moura and Oliveira (2005)	SKP				x			x						x	x	x			x				x				x	x				
		SLOPP				x			x						x	x	x			x				x				x	x				
47	Moura and Oliveira (2009)	SSSCSP	VRPTW			x		x	x			x	x		x	x	x							x				x	x				
48	Mustafee and Bischoff (2013)	SLOPP				x	x	x	x						x				x		x					x							
49	Ngoi et al. (1994)	SKP				x			x						x		x				x					x							
50	Parreño at al. (2008)	SKP				x			x						x			x				x	x					x				x	
		SLOPP				x			x						x			x				x	x					x				x	
51	Parreño et al. (2010b)	SKP				x			x						x			x					x					x					
		SLOPP				x			x						x			x					x					x					
52	Ratcliff and Bischoff (1998)	SLOPP				x	x		x						x		x							x								x	
53	Ren et al. (2011)	SLOPP				x			x		x				x		x							x				x				x	
54	Scheithauer and Terno (1997)	SSSCSP		x		x	x		x						x			x					x								x		
55	Tao and Wang (2013)	SBSBPP	VRP			x	x		x				x		x			x			x						x						
56	Tarantilis et al. (2009)	SBSBPP	VRP	x		x	x	x	x			x	x		x			x			x						x						
57	Techanitisawad and Tangwiwatwong (2004)	SKP			x	x	x		x						x			x		x	x						x						
		MBSBPP			x	x	x		x						x			x		x	x						x						
58	Terno et al. (2000)	SSSCSP		x		x	x	x	x			x	x		x			x					x								x		
59	Yeung and Tang (2005)	ODP				x			x						x				x		x										x		
60	Zhang et al. (2012)	IPP				x			x						x		x							x				x					
		SKP				x			x						x		x							x				x					
		SLOPP				x			x						x		x							x				x					
61	Zhu and Lim (2012)	IPP				x			x						x		x							x				x					
		SKP				x			x						x		x							x				x					
		SLOPP				x			x						x		x							x				x					